Dez./Jan. 1986/87 DM 5.50 ÖS 48 SF 5.50

CK

Die User Zeitung

mit Sonderteil fur Commodore
Atari, Sinclair, TI 99/4A

Nr. 12/1 3. Jahrgang

Computer Kontakt

Atari

- Reviews: The M[illegible] Graphi[illegible] Rep[illegible]
- Kurse: Assemble[illegible] 6502 Ma[illegible] sprache-Ku[illegible]
- Listings: Filec[illegible] Wa[illegible]

Sinclair Spectrum

- [illegible]
- [illegible]
- [illegible]chinensprache-[illegible]
- [illegible]Face

Sinclair QL

- [illegible] + Tricks
- [illegible]oberfläche

TI 99/4A

- T[illegible]
- Hexmonitor in C
- Combas

Commodore

- Vietnam
- Grafik-Kurs Teil 3

QL-Seiten

ATARI ST

ASSEMBLER-PRAXIS
AUF ATARI ST
Roland Löhr
tewi

ASSEMBLER PRAXIS AUF ATARI ST

[illegible] MICRO-MAG [illegible] ATARI ST

Erklärt Grundlagen

[illegible] 68000er [illegible]

Zeigt Anwendungen

[illegible] ATARI ST [illegible] GEMDOS [illegible]

Entwickelt Hilfsprogramme

[illegible]

ca. 300 Seiten, Softcover, DM 59,–

tewi Verlag GmbH
Theo-Prosel-Weg 1
8000 München 40

Weitere te-wi-Bücher

NEU

DAS C-BUCH

[illegible]

UNIX

[illegible] DM [illegible]

LOGO –
Jeder kann programmieren

[illegible] USA [illegible] Papert [illegible] DM 59

M68000 FAMILIE [illegible]

[illegible]

Bd. 1 [illegible]

Bd. 2 [illegible] DM [illegible]

UMWELTDYNAMIK

[illegible] 480 Seiten [illegible] DM [illegible]

Mein ATARI Computer

[illegible]

Spielende Ideen mit ATARI GRAPHIK

[illegible]

6502 – Programmieren in Assembler

[illegible] DM [illegible]

Noch im Programm: Einführung in die Microcomputer-Technik, DM 66,–
Computer für Kinder, ATARI, DM 29,80

VORWORT

Liebe Leser,

eine etwas traurige Mitteilung müssen wir den Commodore-Lesern unter Ihnen machen: Wir werden ab dem nächsten Heft den Commodore-Teil einstellen. Der Grafik-Kurs wird allerdings zu Ende geführt. Sie werden jetzt sicher nach dem Grund fragen. Wie auch bei den Computern selbst findet im Zeitschriftenbereich ebenfalls eine Konzentration statt. Dies hat bei uns dazu geführt, daß wir bei den Computertypen von Sinclair und Atari sowie dem TI 99/4A immer stärker wurden, dagegen bei Commodore gerade in der letzten Zeit Leser verloren haben. Dieser Prozeß wird sich weiter fortsetzen, weshalb wir uns in Zukunft noch stärker auf die restlichen drei konzentrieren wollen. Speziell den QL und den TI werden wir dabei noch mehr berücksichtigen.

Dagegen gibt es für die Atari-Leser eine interessante Neuigkeit. Am 22.12.86 erscheint die erste Ausgabe unseres neuen ATARI magazins. Lassen Sie sich dieses Heft nicht entgehen, auch die kleinen Ataris werden wir dort gebührend berücksichtigen. Neu ist auch unser Atari-Buchversand hier im Heft mit insgesamt 30 Titeln. Da ist für jeden etwas dabei.

Auch wer regelmäßig unsere Software-Cassetten bestellt, wird sich freuen. Ab sofort gibt's die neuen Programm-Cassetten für Atari, Spectrum und TI 99/4A mit den Listings aus Computer Kontakt.

Bis zum nächsten Mal
Ihr

Thomas Ebe

Thomas Eberle, Chefredakteur

Brandheiße Knüllerpreise

TI-99/4 A

[illegible]

Schneider

[illegible]

Atari

[illegible]

Commodore

[illegible]

CSV RIEGERT
[illegible]

Jupitersoft GdbR

Neueste Software für
C64 C128 SVI 318/328
VZ 200 Laser 210 310
ab **7.90 DM**

Adventures Action Anwender
Grafik und Dateiprogramme

Viel Software stark im Preis reduziert.

Kostenloser Katalog bei

Robert Goth
Jupitersoft GdbR
Höttingerstr. 34b
8830 Ellingen
(09141)8174

unsere seit
U&J Rudolph
Versand
Ehlenerstr. 7
3501 HBGF
Verkauf
Druselturm
3500 KASSEL
0561/770367

ATARI ST Tagestiefstpreis
Spectrum ab 240.-
Sinclair QL ab 440.-
Tastatur f QL ab 180.-
Opus Discovery ab 380.-
Seikosha SP 1000 ab 699.-
Computer Drucker + Zubehör in unserer Gratisinfo
Unseren Kunden
ein frohes Weihnachtsfest
Ursula & Joachim Rudolph

INHALTSVERZEICHNIS

RUBRIKEN

Vorwort 3
Topprogramm 5
Clubnachrichten 6
Atari Buchversand 60
CK Programmservice 65
Buchversand 93
Kleinanzeigen 118
Fundgrube, Inserentenverz. u. Impressum 121

USER-CLUB SINCLAIR

ZX Spectrum Maschinensprachekurs 10
Assemblertips für den Spectrum 12
ISO FACE 1.0 16
Textverarbeitungsprogramm 10
Graf Edit — ein Grafikeditor 17
3 kurze Programme für Spectrum 24
Quercopy für Spectrum 16/48 K 26
Zitterschrift 28
Window Routine 30
Reepos für ZX 81 32
Indexomat für den QL 36
Microsoft Basic auf dem QL 38
Grafische Benutzeroberfläche für den QL 36
Tips und Tricks für den QL 40

COMMODORE-NEWS

Tips und Tricks zu C 64 und Floppy 1514 46
Grafikkurs C 64, Teil 3 47
Vietnam für den C 64 51
Input de Luxe 55
Amiga Spieleübersicht 57
Lattice auf Amiga-Pfaden 58
Kurs: [illegible] Makro-Assembler 59

ATARI

Graphic Arts Department 62
Neues für den Gabentisch 64
Kurs 6502 — Maschinensprache 68
Pabers Assembler Ecke 74
Spiel zum Abtippen: Wombel 78
Disk Editor 81
Kalkulationsprogramm Calc 400 82
Hardcopy auf Seikosha SP 500 AT 88
Speed Tape für Atari XL/XE 89
Filecopy 90
Zeichensatzfinder 91

ATARI ST

Profi Painter 95
Software-Test: ST [illegible] 97

TI 99/4A

Mechatronic-Eprommer [illegible] 101
Neue Software: TI Artist 102
Neues von Turbo-Assembler 103
Lange Programme speichern 104
C auf dem TI 99/4A 106
Hexmonitor 107
Spiellisting SR302 109
U-Boot-Schlacht 112

Bei uns können Sie mitmachen

Computer-Kontakt ist die Homecomputerzeitung zum Mitmachen. Sie können bei uns Programme einsenden, Bücher besprechen, Spiele beschreiben, Tips und Tricks schicken, Fragen stellen und Ihre Meinung sagen. Wir haben für alles ein offenes Ohr. Damit wir aber Ihre Einsendung schnell bearbeiten können und alles mit rechten Dingen zugeht, müssen Sie folgende Punkte beachten:

1. Ihr Brief sollte ein Anschreiben mit Name, Anschrift, Telefon und Einsendedatum enthalten.
2. Geben Sie genau an, welches Gerät Sie haben. Läuft das Programm nur mit Speichererweiterungen oder Zusatzgeräten, müssen diese unbedingt angegeben werden.
3. Zu jedem Programm sollte eine Programmbeschreibung beiliegen. Diese kann mit der Schreibmaschine oder mit einem Drucker geschrieben sein. Der Zeilenabstand muß 2 Zeilen betragen, damit noch Korrekturen oder Anmerkungen eingefügt werden können.
4. Zu jedem Programm gehört grundsätzlich ein Listing und eine Cassette oder Diskette. Wenn Sie aber keinen Drucker haben, reicht auch der Datenträger. Speichern Sie zur Sicherheit das Programm zweimal ab. Cassetten und Disketten können wir nur zurücksenden, wenn Rückporto beiliegt.
5. Berichte, Spielebeschreibungen und Buchbesprechungen müssen ebenfalls zweizeilig geschrieben werden.
6. Wenn wir ein Programm von Ihnen abdrucken, vergüten wir ein Honorar für den einmaligen Abdruck und die Nutzung des Programms in unserem Cassettenservice. Die Höhe des Honorars richtet sich nach der Länge und Qualität des Programms. Wir vergüten im allgemeinen bis zu 300 DM, für sehr gute Programme kann es auch mehr sein.
7. Mit der Einsendung gibt der Verfasser die Zustimmung zum Abdruck und erklärt, daß er Urheber der Texte und Programme ist und das uneingeschränkte Nutzungsrecht daran besitzt. Sollte der Einsender Programme einschicken, an denen er kein Urheberrecht und kein Nutzungsrecht besitzt, hat er bei Abdruck durch uns etwaige Schadensersatzansprüche von seiten Dritter selbst zu tragen.

Die nächste Ausgabe »Computer Kontakt« erscheint am 27.1.1987

Topprogramm "Graf Eddi" für den Spectrum

Mit "Graf Eddi" lassen sich vorhandene Grafiken spiegeln drehen invertieren verschieben kopieren und vieles mehr Auch eine 24-Zeilen-Hardcopy ist möglich

Der Autor des Programms, Stefan Arbes, ist 17 Jahre alt und besucht das Friedrich Dessauer Gymnasium in Aschaffenburg Neben der Programmierung von Rechnern beschäftigt er sich mit der hardwaremäßigen Anwendung von Computern bei der Steuerung im Bereich der Musik- und Bühnenelektronik

Computer-Kontakt das Heft mit den preisgünstigen Kleinanzeigen

1000 DM Honorar

In der CK gibt es das Toplisting Das läuft so, daß wir hier in der Redaktion von allen Programmeinsendungen das beste Programm heraussuchen und in der CK als Toplisting abdrucken Der Autor dieses Programms erhält dann als Honorar 1000 – DM

Mitmachen können alle Programmierer mit den Geräten Atari, Sinclair und TI 99/4A Ein Listing muß nicht unbedingt beiliegen, falls der Autor des Programms noch keinen Drucker hat Werden Programme abgedruckt, die nicht zum Toplisting gewählt werden, erhält der Autor dafür ganz normal das übliche Honorar Beim Toplisting ist das Honorar für den Abdruck in den 1000 – DM schon enthalten Mit der Einsendung seines Programms erklärt sich jeder Autor mit den einzelnen Punkten im Text "Bei uns können Sie mitmachen" einverstanden (siehe Seite 4 gegenüber)

Deshalb Leute aufgepaßt Bei uns kann man Geld verdienen Die Chancen für einen Abdruck, oder gar Gewinner des Toplistings zu werden, stehen bei uns immer gut

Abo-Bestellschein

Ich möchte Computer Kontakt in Zukunft regelmäßig zugeschickt bekommen Meine Abo-Bestellung gilt ab der nächsten Ausgabe Die Abodauer beträgt 6 Ausgaben also ein Jahr und kann bis spätestens 4 Wochen vor Aboende wieder gekündigt werden Der Abonnementpreis beträgt 33 DM einschließlich Mehrwertsteuer und Versandkosten. Für Bestellungen aus dem Ausland wird es aber nur ein wenig teurer Hier kostet das Abo 37 50 DM

Name/Vorname

Straße PLZ Ort

Ich bezahle wie folgt Ich bestelle ab Ausgabe

☐ Scheck liegt bei

☐ Vorauskasse auf Postscheckkonto Karlsruhe Nr 43473 756

Mir ist bekannt, daß ich diese Bestellung innerhalb 8 Tagen widerrufen kann und bestätige dies mit meiner Unterschrift (Dieses Wiederrufsrecht ist per Gesetz vorgeschrieben)

Datum/Unterschrift

Diesen Bestellschein ausschneiden oder fotokopieren an Computer Kontakt Postfach [illegible] Mittelstenscheiden

Der Jeposoft-Preishammer hat zugeschlagen!

QL (engl.) mit 640K Speicher **DM 979.00**

QL (engl.) mit 128K Speicher **DM 444.00**

Vorführung und Selbstabholung der Ware nur nach tel. Vereinbarung

evtl. Leasing (Mietkauf)

Abdeckhaube für QL **DM 34.00**

Abdeckhaube für Spectrum+ **DM 19.90**

MICRODRIVE CARTRIDGE STORAGE BOX inklusive 20 cartridges

DM 18.50

4 Cartridges DM 30.-

Cartridge-Box mit 20 Cartridges **DM 169.50**

Multiface One neue Version **DM 186.00**

Brandneu: Spectrum-Software

Der Sturm des Druden deutsches Adventure **DM 49.90**

JEPOSOFT

Kruppstraße 9, 4040 Neuss 21

telefonisch erreichbar ☎ 0 21 07/81 84

USER-CLUBS

Atari-Clabs

Unser Atari-Club sucht noch Mitglieder. Wir sind derzeit 19 User, die entweder einen 600/800 XL bzw. einen 130 XE besitzen. Bei genügend großer Mitgliederzahl ist eine monatlich erscheinende Club-Zeitung geplant. Ansonsten bieten wir eine Soft- und Hardware-Vermittlung an, auch eine eigene Software-Bibliothek ist vorhanden. Wir klären auch Fragen von Mitgliedern. Info gegen 80 Pf Rückporto. Anfragen aus dem ganzen Bundesgebiet sind willkommen!

Michael Scholz
Adlerstr. 42
2300 Kiel 1

Unser Atari-Club besteht seit 3 CK-Ausgaben. Wir beschäftigen uns mit den Atari Computern 400/800 XL und XE. Programmiert wird in Maschinensprache und Basic. Anfängern und Fortgeschrittenen stehen wir mit Rat und Tat zur Seite. Jedes Clubmitglied erhält alle 3 Monate die Clubdiskette mit Utilities, Anwendungen, Spielen, dem Maschinensprachekurs und dem Clubmagazin. Weitere Infos gibt es bei

Damata
D. Boonstra
Am Schierbrunnen 15
2120 Lüneburg

Wir einige Atari-User aus Wuppertal haben einen Computer Club gegründet und suchen noch weitere Mitglieder (600 XL - 130 XE). Ein Clubmagazin auf Diskette ist in Vorbereitung, es soll zum Selbstkostenpreis (ca. 5,- DM + Porto und Verpackung) abgegeben werden. Einen Clubbeitrag wollen wir möglichst nicht erheben, aber eventuelle Unkosten, die bei einem Treffen aufkommen (z.B. Getränke). Anfragen von Raubkopierern sind zwecklos. Info gegen Rückporto.

ACCW
Gino Wintermantel
Gemarker Ufer 15
5600 Wuppertal 2

Hallo Atari-User!

Wir wollen im Raum GF, WOB, BS, SZ und WF einen Computer Club gründen. Auch eine Club-Bibliothek sowie Club-Treffs mit Unterhaltung und Erfahrungsaustausch sind geplant.

Oliver Hamann
Gladiolenweg 2
3170 Gifhorn

Mein Freund und ich sind beide 14 Jahre alt und suchen Kontakt zu anderen Atari-800-XL-Besitzern mit Datasette oder auch Floppy. Sie sollten in unserem Alter sein und auch im Raum Duisburg wohnen.

Daniel Domenach
Sandstraße 3
4100 Duisburg 11

Kontakt gesucht!

Ich suche Kontakt zu Spectrum-Usern in der Gegend des Dümmer Sees (auch Urlauber).

Michael Bosse
Jadingen 454
4995 Stemwede 2

Ich möchte einen Sinclair Club gründen, der sich mit dem 128er beschäftigen soll. Ein Schwerpunkt der Clubarbeit wird die Software sein. Wer Interesse hat, kann sich melden.

Sebastian B. [illegible]
Schillerstraße 9
5444 Bad Breisig
Tel. [illegible]

Atari-Club

Wir sind vermutlich ein Club mit den gleichen Interessen wie alle anderen und suchen Gleichgesinnte im Raum Lippstadt, Rietberg, Mastholte und Umgebung. Wir besitzen Atari 400 bis 130 XE mit 16 K, Basic, Diskette, Cassette, einen Epson FX 80 und eine umfangreiche Bibliothek mit jeder Art von Programmen. Wir suchen keine Leute, die postlagernd arbeiten und nur Programme abstauben wollen. Wir erheben keinen Club-Beitrag und keine Portogebühren. Wir wollen Programme tauschen, Probleme lösen und Tips und Tricks weitergeben. Bei uns gibt es keine Altersgrenze. Für uns sind die Freaks richtig, die mal schnell mit dem Fahrrad oder dem Auto zusammenkommen. (Abholung ist auch möglich.)

Martin Schiller
Schulstraße 5b
4835 Rietberg 3/Mastholte

Fühlt auch Ihr Euch in Bezug auf Computerliteratur für Euren Spectrum immer mehr vernachlässigt??

Seid Ihr daher auch auf der Suche nach einer optimal abgestimmten Zusammenstellung von Tips & Tricks, Programmlistings, Bauanleitungen und Erfahrungsberichten bezüglich dieses Homecomputers???

S O N D E R H E F T

JA?

Dann haben wir vom Spectrum-Profi-Club hier genau das Richtige für Euch, nämlich das aktuelle "Spectrum Profi Sonderheft". Es handelt sich hierbei um ein seiten- und inhaltsstarkes Heft, dessen Inhalt optimal abgestimmt wurde. Dazu wurden die besten Artikel aus unseren sämtlichen, bisher erschienenen Clubmagazinen genommen und hier zusammengestellt. Wir haben uns bemüht, möglichst viele Anwendungsbereiche zu treffen und zu behandeln, um dadurch ein umfangreiches Informationsangebot zu schaffen. Von Interface-Bauplänen über ernsthafte Anwenderlistings bis zum Poke, der den Spectrum bei Profisituationen in die ewigen [illegible] Jagdgründe verweist, ist wirklich fast alles vertreten. Dieses "Spectrum-Profi-Sonderheft" kann gegen Erstattung unserer Unkosten mit 10 DM bei uns angefordert werden (per Scheck oder per verbindlicher NN-Bestellung, dann jedoch 15 DM). Spectrum Profi-Club Lärchenstr. 2 8901 [illegible]

Wer sich für den Club selbst und dessen Leistungen interessiert, kann sich gegen Beifügung weiterer 5 DM Unkostenerstattung noch umfangreiches Infomaterial und ein Probeexemplar des Clubmagazins zukommen lassen. Nun also als kleiner Querschnitt einige Ausschnitte aus dem eigentlichen Inhalt des Sonderheftes. ● Bauplan einer Centronics- und Kempston-Schnittstelle mit passender Software ● Auflistung von ROM-Routinen, Systempokes und Spielepokes ● Ferner jede Menge Basic- und MC-Listings ● Tips und Tricks in jeder Lebenslage ● Computerkurse ● Erfahrungsberichte ● Tests und so weiter...

★ Business-Software ★
★ Commodore – Sinclair ★
★ QL-Software & Zubehör ★

PC-FIBU ab 199,–
PC-SM Business [illegible] 398,–
C64-FIBU ab 98,–
C64 Kunden-/Lief. Buchh. 98,–
Vizawrite-Vizastar Multiplan
SP Tasword II 49,50
SP Masterfile/MF-Print 89,–
SP Statistik II 49,–
SP [illegible]-Compiler je 98,–
Kempston-E Interface 148,–

Katalog DM 3,–
(Bitte Rechner angeben)

Fa. Lücker/CK
R.-Wagner-Straße 71
6239 Kriftel

Gewußt wo!?

Software + Hardware
– nur Markenprodukte –

– Atari
– Commodore
– Sinclair/Spectrum
– Schneider

Testen Sie uns!
Sie werden erstaunt sein!
Ihre Preisliste liegt bereit!

Annette Lange
Pf 103771 ● ☎ 04241/4862
D-2800 Bremen

DATENKASSETTEN

DISKETTEN

Deutsches Markenband
alle Größen von C- [illegible] lieferbar.
z.B. C [illegible] ab DM 0,[illegible]–DM 0,[illegible]
Copy-Service.
Laufend Sonderangebote, interessant auch für Wiederverkäufer.
[illegible]

3,5" Atari [illegible]
Preis auf Anfrage
5¼" HC/PC Preis auf Anfrage
10 Disketten 5,25" HC/PC
in Plastikbox (transparent o. farbig) Preis auf Anfrage
[illegible]
[illegible] auf Anfrage

Holzschuh Tasso
Keltenstr. 57 [illegible] Bensheim
Tel. [illegible]

LOGICAL-GEWINNER

Bei unserem etwas mißglückten Logical aus Heft 8/1986 haben wir ja alle Einsendungen für die Auslosung genommen, da es keine eindeutig richtige Lösung gab. Hier die Gewinner, die je ein Jahresabonnement Computer Kontakt erhalten.

- Martin Grunz
  Forstwald 3
  8340 Schönau
- Albert Krause
  Auf dem Kohl 62
  5800 Hagen 1
- Wolfgang Lohus
  [illegible] 29
  7000 Stuttgart 40
- Robert Schuh
  Im Lochgarten 11
  5000 Köln 90
- Karin Wimmer
  Schöpperweg 8
  8458 Sulzbach-Rosenberg
- Ludwig Philipp
  Bergangerstr. 17
  8012 Ottobrunn
- Peter Stock
  Pfarrweiherstr. 2
  8590 Marktredwitz
- Ute Ellermann
  Weidenweg 1
  2126 Adendorf
- Harald von Gurkom
  Horner Str. 12
  2221 Tettenhusen
- Markus Wolf
  Buchenweg 5
  3305 Sickte

Wer sucht noch alte CK-Hefte?

Alle neuen Leser haben bei uns die Möglichkeit, die zurückliegenden Hefte nachzubestellen. Die Ausgaben von 1984 sind nicht mehr lieferbar. Bestellt wird mit untenstehendem Bestellschein. Die Lieferung erfolgt aber nur gegen Vorauskasse in Form von Briefmarken oder gegen Scheck.

Bestellschein für CK-Hefte

Ich möchte folgende **CK-Hefte** bestellen

Ex. Heft Januar 85 (4,50 DM)
Ex. Heft März (4,50 DM)
Ex. Heft April (4,50 DM)
Ex. Heft Mai (4,50 DM)
Ex. Heft Juni (4,50 DM)
Ex. Heft Juli (4,50 DM)
Ex. Heft August-September (4,50 DM)
Ex. Heft Oktober (4,50 DM)
Ex. Heft November (4,50 DM)
Ex. Heft Dezember-Januar 85/86 (5,50 DM)
Ex. Heft Februar-März (5,50 DM)
Ex. Heft April-Mai (5,50 DM)
Ex. Heft Juni-Juli (5,50 DM)
Ex. Heft August-September (5,50 DM)
Ex. Heft Oktober/November (5,50 DM)
Versandkosten (1-2 Hefte 1,40 DM
2-4 Hefte 2,00 DM 5-15 Hefte 3,00 DM)

Summe

Meine Anschrift

Den Bestellschein einsenden an den Computer Kontakt, Postfach 1640, 7518 Bretten

Verlangen
Sie mehr!

Rainer Fischer
Datenverarbeitung mit dem Sinclair QL
1986, 178 S., kart., DM 42,–
ISBN 3-7785-1329-7
Das Buch zeigt zunächst die entsprechenden Möglichkeiten von SuperBASIC und führt den Leser dann in die Handhabung der Dateiverwaltungssprache von QL ARCHIVE ein. Der Leser erhält dabei wichtige Hinweise für die Erstellung wohlstrukturierter Programme.

Hajo Lemcke
Assembler mit dem C-64
1985, 227 S., 22 Abb., 8 Tab., kart., DM 38,–
ISBN 3-7785-1091-6
In der Maschinensprache nutzen Sie die Fähigkeiten Ihres Computers erst richtig aus. Diese zweibändige Einführung verhilft Ihnen dazu. Jeder Befehl bzw. jede Gruppe wird mit einem lauffähigen Beispiel vorgestellt, das sich später in größere Programme einbauen läßt.

Thomas Tai
Cracker, Hacker, Datensammler
1986, 98 S., kart., DM 24,–
ISBN 3-7785-1256-8
Die Praktiken der meist jugendlichen Kopierer steht im Mittelpunkt dieser mit vielen authentischen Fällen versehenen Schrift, die in einer Art Bestandsaufnahme des Softwareschwarzmarktes beide Seiten zu Wort kommen läßt.

BESTELLCOUPON
Absenden an: Dr. Alfred Hüthig Verlag
Im Weiher 10, 6900 Heidelberg

Titel
Name, Vorname
Straße, Nr.
PLZ, Ort
Datum, Unterschrift

Hüthig

USER CLUB
sinclair

Sinclair ist ein eingetragenes Warenzeichen der Sinclair Ltd

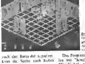
Hallo Freunde,

noch immer ist nicht geklärt, ob und wenn ja, wie es mit Sinclair-Produkten weitergeht. Im Sinclair-Teil dieser CK werdet Ihr z. B. eine Kurznachricht über Thor, den neuen QL-kompatiblen Computer finden. Aus England kommen Meldungen und Gerüchte, wonach es bereits zwei neue Spectrum geben soll. Der eine aus dem Hause Amstrad sieht aus wie der 464 von Schneider, also mit vernünftiger Tastatur und eingebautem Recorder. Von dem anderen Gerät hört man die wundersamsten Dinge, es soll angeblich Fähigkeiten des Amiga erreichen. Wie gesagt, die Gerüchteküche brodelt, Fakten liegen nicht vor.

Im Vorwort der letzten Ausgabe habe ich an dieser Stelle das Programm „G-Force" angekündigt. Wie ihr sicher bemerkt habt, haben wir es dann kurzfristig streichen müssen. Da "G-Force" Überlänge hat, wollen wir es auch jetzt nicht abdrucken. Ihr könnt das Programm aber preiswert direkt auf Cassette beim Verlag bestellen.

Jetzt aber zu den harten Tatsachen. Wir haben – wie üblich für den ZX 81, den Spectrum und den QL wieder eine Menge Informationen aufbereitet. Besonders erfreulich ist, daß das Topprogramm des Monats wie der einmal für den Spectrum geschrieben wurde. Der Grafikeditor in diesem Heft wird sicher vielen Anwendern Freude bereiten. Ähnlich begeistert werden sicher die immer noch zahlreichen Besitzer eines ZX 81 über "Respos" sein. Auf jeden Fall ist der Sinclair-Teil wieder prall gefüllt, die anstehenden Feiertage dürften damit gerettet sein.

Rolf Knorre



Batman
Grafisch aufwendig gestaltet und in der Aufgabe anspruchsvoll

Batman

Robin, der Freund und Helfer Batmans, ist entführt worden und natürlich will ihn der Held sofort befreien. Dazu muß er aber erst einmal die sieben versteckten Teile seines Batmobils finden. Laut Anleitung sind rund 150 Räume zu durchsuchen, in denen es nur von Gefahren nur so wimmelt. Bei Programmstart ist der Held schlecht ausgerüstet. Für eine erfolgreiche Suche müssen nun außer den Fahrzeugteilen noch andere Ausrüstungsgegenstände aufgetrieben werden. Die Batboots z. B. ermöglichen Batman das Überspringen diverser Gefahren. Weitere Extragaben machen Batman schneller oder vorübergehend unverwundbar, sogar ein Extraleben kann erhalten. Mit der kompletten Ausrüstung und den Zugaben steigen Batmans bzw. des Spielers Chancen ungemein. Ist auch das Batmobil repariert, kann die Suche nach Robin endlich beginnen.

Bekanntlich sind neben Kinohits auch Comic-Helden beliebtes Material für Programmierer bzw. für Softwarefirmen, die sich allein durch den bekannten Namen schon ein Geschäft versprechen. Jüngstes Opfer dieses Trends ist Batman, der Held von Gotham City, der schon seit Jahrzehnten mit seinem immerjungen Helfer Robin in dieser Stadt für Ordnung sorgt. Bei der Bearbeitung durch Ocean entstand ein Action-Adventure, das sich hauptsächlich auf das schon übliche Sammeln und Suchen diverser Gegenstände konzentriert.

Das Programm ist am Spielen wie "Knight Lore" oder "Gunfright" zu vergleichen, auch hier ist eine 3-D-ähnliche Animation vorhanden. Da es mittlerweile Unmengen solcher Programme gibt, ist es sehr schade, daß den Programmierern zu diesem Thema nicht mehr eingefallen ist. Es muß aber auch gesagt werden, daß "Batman" grafisch sehr liebevoll gestaltet wurde und die gestellte Aufgabe so kniffig ist, daß man längere Zeit benötigt, um damit klarzukommen. Alles in allem ein nettes Programm, dem eine neue Spielidee aber gutgetan hätte.

System ZX Spectrum 48 K
Hersteller Ocean

Rolf Knorre

Paperboy

Der deutsche Vertrieb dieses Programms (Fuler West Records) hat uns zum Test eine Vorabversion von "Paperboy" zur Verfügung gestellt. Mit diesen Versionen ist das immer so eine Sache, da man nie genau weiß, was noch fehlt. Ich glaube aber, daß Paperboy bis auf den noch fehlenden Sound schon fast fertig ist. Wenn ihr dieses Heft in Handen haltet soll das Programm aber bereits in der endgültigen Version vorliegen.

Worum geht es nun bei "Paperboy"? Wer hin und wieder Spielhallen besucht, weiß sicher schon Bescheid, denn dort gibt es das Spiel schon länger. Es hat sich sogar zu einem echten Renner entwickelt. Hier kann ich gleich einmal erwähnen, daß die Umsetzung auf den Spectrum den Programmierern hervorragend gelungen ist, die Kopie steht dem Original kaum nach.

Die Spielhandlung ist recht simpel. Nach amerikanischem Vorbild muß ein kleiner Junge auf seinem Fahrrad Zeitungen ausliefern, d.h. die Zeitungen im Vorbeifahren in die entsprechenden Kästen (oder daneben) werfen. Dafür gibt es Punkte. In seiner Tasche hat der Paperboy jeweils 10 Exemplare. Sind diese verteilt, muß er vom Straßenrand ein neues Paket aufnehmen. Das klingt hier ziemlich einfach, erweist sich in der Praxis aber als wesentlich schwieriger. Der Bürgersteig, auf dem der kleine Kerl entlangsaust, ist nämlich nicht nur für ihn da. Mal kommen ihm Fußgänger oder Tiere entgegen, mal steht eine Mülltonne im Weg. Besonders haarig wird es, wenn er die sehr verkehrsreiche Straße überquert. Es gibt jedenfalls zahlreiche Überraschungen, die ich hier nicht alle verraten will. Das Spiel ist nicht in Level, sondern in Wochentage unterteilt. Nach jedem Tag bekommt der Spieler einen Service Report auf den Bildschirm, bevor es mit dem Streß weitergeht.

"Paperboy" bietet eine witzige Spielidee und -umsetzung und eine tolle Grafik. Mehr kann man kaum verlangen. Dieses Programm sollte bei keinem spielbegeisterten Spectrum Freak in der Sammlung fehlen.

System	Spectrum 48 K
Hersteller	Elite

Rolf Knorre

Hocus Focus

Jeremy, hauptberuflich Reporter, nebenberuflich Punk, soll für seinen Arbeitgeber, den Daily Shocker, eine Story recherchieren. Irgendein Professor hat eine tolle Erfindung gemacht, bei der es um Mutationen geht. Soweit die Rahmenhandlung zum Programm "Hocus Focus". Klingt doch eigentlich ganz interessant. Die Umsetzung auf den Spectrum läßt davon aber nicht mehr viel übrig. Auf dem Bildschirm stellt sich "Hocus Focus" als übliches Lauf- und Suchspiel dar. Grafik und Sound liegen aber unter dem Durchschnitt. Man kann sich mit diesem Programm eine Weile beschäftigen, kann es aber genausogut bleibenlassen.

System	Spectrum 48 K
Hersteller	Quicksilva

Rolf Knorre

Hocus Focus: Lauf und Sucharbeit eines Reporters.

★ ZX-Spectrum ★ ZX-Spectrum ★ ZX-Spectrum ★

DISCOVERY 1(0) DM 398,–
SANYO-MONITOR (z.B. f. Discovery) DM 260,–
PANASONIC-Drucker KX P 1090 DM 842,–
KEMPSTON-E-Drucker Interface DM 149,–
Tastatur (dk'tronics) DM 118,–
dk'tronics-Drucker Interface DM 95,–
Trojan-Lightpen DM 59,–
48 k-Erweiterung DM 49,–
Joystickinterface DM 29,–
GAMES PLAYER Joystickinterface mit Regelung der Spielgeschwindigkeit DM 99,–
SPRACHSYNTHESIZER dk'tronics DM 79,–
3-KANAL-SOUND-SYNTHESIZER DM 79,–
48-K Spectrum Kundendienst rückläufer mit kleinen Fehlern ... DM 148

Liste U.K. mit weiteren Angeboten anfordern oder direkt bestellen bei:

U. Kunz Junge Hälden 3
D-7500 Karlsruhe 41
Telefonische Bestellannahme 0721/48 28 76 (bis 20 Uhr)

New's New's New's Ständig die neuesten Programme für alle gängigen Rechner (Commodore C16, C64/128, Amiga, CPC, Sinclair, Atari, ST)

New's Softwarevertrieb
Inhaber Karl-Heinz Klug
Händler-Tel. 0211/489530

Michael Naujoks

● Spectrum ● Spectrum ● Spectrum ● Spectrum ● Spectrum ●

Light Pen DM 66,–
3-Kanal Soundsynthesizer DM 95,–
Speechsynthesizer DM 69,–
Spectrum DM 149,–
Multiface One DM 139,–
Joystick Interface DM 29,–
Joystick Turbo Pro DM 39,95
The Art Studio DM 69,–
Tasword II DM 29,–
Masterfile DM 69,–
1942 DM 25,90
Dan Dare DM 32,90
Dragons Lair DM 32,90
Glider Rider DM 30,90
Great Escape DM 25,90
Heart Land DM 34,90
Knight Rider DM 29,–
Moonlight Madness DM 29,90
Nightmare Rally DM 27,90
Now Games 3 DM 32,90
Paperboy DM 29,–
Revolution DM 25,90
Street Hawk DM 25,90
The Trap Door DM 25,90

Hardware Sinclair Atari Software

Spectrum 128 K
Spectrum 128 K DM 4..,–
Asterix DM 34,90
Fairlight DM 34,90
Hacker DM 34,90
Nodes of Yesod DM 34,90
The Planets DM 34,90
QL
QL DM 4..,–
QL Art DM 95,–
Cartridge Doctor DM 59,90
Mort Ville Manor DM 69,90
Othello DM 42,90
Wanderer DM 59,90
Atari ST
Electronic Pool DM 59,90
Hacker II DM 79,90
Karate DM 69,90
Leaderboard Tournament DM 37,90
Mercenary DM 69,90
Protector DM 69,90
PSION Schach DM 79,90
Shuttle II DM 69,90
Fight Winder DM 64,90
Space Station DM 69,90
Star Glider DM 79,90
Temple of Apshai DM 69,90
Wintergames DM 79,90

Kostenlosen Katalog S/10 anfordern!

Alle Preise zuzügl. Porto ... bei Nachnahme ... Unbedingt Computer-Typ angeben.

Entwicklung & Vertrieb von Computer Soft- und Hardware
Rohrbacherstr. 40 6900 Heidelberg
Hotline (06221) 46886

Maschinensprachekurs auf dem ZX-Spectrum

Teil 4 Programmaufbau Ein- und Ausgabebefehle

Nachdem in den letzten Teilen dieser Serie schon verschiedene Befehlsgruppen erklärt wurden, wird diesmal schwerpunktmäßig auf die eigentliche Entwicklung eines MC-Programms eingegangen. So ganz nebenbei werden dabei noch die Ein- und Ausgabebefehle der Maschinensprache erläutert.

Wenn man sich an seinen Spectrum setzt, um ein MC-Programm zu schreiben, sollte man sich erst einmal auf einem Blatt Papier notieren, wie man den zur Verfügung stehenden Speicherplatz nutzen möchte. Dazu nimmt man folgende Werte zu Hilfe: Die Größe des Basic-Programmteils inkl. Variablen und möglicher Microdrive-Maps und Kanaldaten ist nötig, um den Beginn des freien Speicherbereichs festzustellen. Dies kann man, nachdem alle Variablen und Kanäle belegt worden sind, durch Abfrage der Systemvariablen STKEND (Peek 23653 + Peek 23654 *256) relativ genau feststellen. Man sollte dann noch ca. 1 KByte dazuzählen, da noch der Bereich für den Stack und GOSUB-Stapel folgt (siehe auch Speicheraufteilung im Spectrum-Handbuch, Kapitel 24). An dieser Adresse kann man nun mit dem Basic-Befehl CLEAR >Adresse< RAMTOP setzen, damit der Basic-Teil seine genauen Grenzen kennt. Der freie Speicherbereich für das MC-Programm beginnt nun genau 1 Byte über RAMTOP.

Um das Programm schreiben zu können, wird natürlich noch ein Assembler benötigt. Je nach Art handelt es sich dabei um einen ortsfesten (steht immer an der gleichen Stelle im Speicher) oder frei verschiebbaren Assembler. Aus der Anleitung sind nun seine Länge und (bei ortsfesten) die Startadresse zu entnehmen, um den Platz bei der Speichervergabe zu berücksichtigen. Des weiteren ist noch beim Sourcetext des Assemblers zu beachten, daß es sehr viel Platz verbraucht (auf 1 KByte Code kommen ca. 8 KByte Source, bei viel Kommentar sogar einiges mehr). Dies ist aber nur dann von Bedeutung, wenn man große Programme am Stück schreibt und diese mit geladenem Assembler testen will. So dürfte es normalerweise keine Speicherprobleme geben, da die meisten Assembler die verschiedenen Blöcke des freien Speichers immer auf deren Sicherheit vor gegenseitigem Überschreiben prüfen.

Hat man nun einen Platz für sein Programm gefunden, so kann das eigentliche Programmieren beginnen. Ein Programm besteht zumeist aus lauter kleinen, vorher einzeln getesteten Routinen, die dann von der Hauptroutine des Programms je nach Bedarf aufgerufen werden. Jede dieser Routinen besitzt verschiedene Merkmale: Welche Übergabeparameter stehen in welchen Registern oder werden aus welchen Speicherbytes geholt? Welche Register und/oder Speicherbytes werden von der Routine verändert und/oder enthalten ein mögliches Ergebnis? Daneben können noch Länge und Ausführungszeit der Routine für das Gesamtprogramm von Bedeutung sein. Mit dieser Methode kann man nun eigenständige Unterprogramme schreiben, die auch in anderen Programmen ohne Probleme verwendet werden können.

Daneben benötigt ein MC-Programm noch verschiedene Teile des Speichers für Daten. In diesen Bytes sind dann z.B. Daten über den Aufbau einer im Programm vorkommenden Figur und Programmvariablen (z.B. aktuelle Position der Figur) abgelegt. Dabei sollte man aber nicht vergessen, daß im ROM des Spectrum schon umfangreiche Daten zur Verfügung stehen (z.B. der Zeichensatz ab Adresse 3D00h bis 3FFFh). Man kann also von seinem eigenen MC-Programm auf Routinen im Spectrum-ROM zugreifen, sollte sich aber vorher ein dokumentiertes ROM-Listing über den Buchhandel (z.B. "Das Sinclair Spectrum ROM", Hueber Verlag) besorgen, damit man auch weiß, welche Routinen vorhanden sind und wie diese arbeiten.

Das abgedruckte Listing ist ein Beispiel für den Aufbau eines einfachen MC-Programms. Begonnen wird mit der Hauptaufrufleiste für die jeweiligen Unterprogramme. Diese sind mit ihren Beschreibungen hinter der Hauptroutine abgelegt. Am Schluß folgen noch die definierten Bytes mit den Programmdaten. Die Hauptsprungleiste des Programms, beginnend bei Label START, springt nach Beendigung aller UPs wieder an den Start der Routine. Damit man diese kleine Endlosschleife aber auch wieder unterbrechen kann, wird zu Beginn erst einmal eine BREAK-Routine aufgerufen, die über eine ROM-Einsprungadresse die Shift- und Space-Taste abfragt und das Ergebnis auswertet. Dazu wird noch ein Speicherbyte des Programms (Label RETURN) auf einen bestimmten Wert hin überprüft. Ist dieser vorhanden oder BREAK gedrückt, wird das MC-Programm nach Aktualisieren des Stacks verlassen. Das Label RETURN kann auch von einem anderen Programmteil aus gesetzt werden, um ein Beenden des Programms zu signalisieren. (Dies geschieht in diesem Beispiel bei der STEUER-Routine.) Eine solche BREAK-Abfrage sollte man an mehreren Stellen eines umfangreichen Programms einbauen, um vorkommende Endlosschleifen elegant zu unterbrechen.

Die zweite Routine, welche von der Endlosschleife aufgerufen wird, ist das PRINT-Unterprogramm. Es holt aus der durch das Label AENDER gekennzeichneten Speicherstelle den Wert und vergleicht ihn auf einen bestimmten Zustand (44h), der dem UP signalisiert, daß sich die Position der Figur geändert hat und neu gezeichnet werden muß (Zeilen 300-420). Dies ist ein Beispiel für die Übergabe von Ergebnissen an andere UPs, hier von der STEUER- an die PRINT-Routine. Nachdem auf diese Weise festgestellt wurde, daß sich die Position der Figur verändert hat, wird diese ab Label PRINT1 gezeichnet. Dazu wird aus der in BC stehenden Position im Unterprogramm UMWAND die Bildschirmadresse mit Hilfe einer weiteren ROM-Routine errechnet. Diese Routine gibt in HL die dazugehörige Bildschirmadresse und in A die Position des Bits in dieser Adresse zurück. Nun weiß unser Programm, an welcher Stelle im Byte die Figur beginnt, also wie oft diese verschoben werden muß (Zeilen 490-510). Danach werden die ermittelten zwei Bytes an die vorbestimmten Stellen im Bildschirmspeicher geschrieben. Dies geschieht durch eine XOR-Verknüpfung (Zeilen 520-570), wodurch die Figur bei wiederholtem Schreiben an der gleichen Position gelöscht (Zeilen 550-560) und an die neue geschrieben wird. Der in Zeile 570 verwendete Befehl ist ein üblicher Trick, um das A-Register auf 0 zu setzen.

Der letzte Programmaufruf der Hauptsprungleiste vor einem erneuten Durchlauf der Schleife wird zur Abfrage der Cursor-Tasten 5 bis 8 und 0 benötigt. Das Unterprogramm STEUER verwendet dazu den für uns neuen Befehl zur Abfrage von Ports. Dies sind die Verbindungen der Zentraleinheit mit der Peripherie (Tastatur, Lautsprecher, Drucker, Microdrive). Für die Eingabe (von der Peripherie an den Rechner) wird der Befehl IN Zielregister (Adresse) verwendet. (Adresse bezieht sich hier nicht auf ein Speicherbyte, sondern auf die Adresse, unter der das abgefragte Gerät an den Rechner angeschlossen ist.) Beim Spectrum ist die eigene Peripherie sogar nur vom Zustand einzelner Bits der Adreßzahl abhängig (siehe auch die Tabelle der wichtigsten IN- und OUT-Adressen). Bei der Abfrage gibt es noch eine Unterscheidung zwischen den Befehlen IN A,(C) und IN A,(wert). Beim 2. Befehl gibt der Wert zwischen den Klammern die Adresse des Gerätes an, also eine 8-Bit-Adresse. Beim 1. Befehl ist aber neben dem C-Register auch noch das B-Register für die Adresse zuständig, so daß sich aus dem Wert des Registerpaares BC eine 16-Bit-Adresse des Gerätes bildet. Dies ist z.B. bei der Abfrage der Tastatur-Ports ausschlaggebend. Der Inhalt des abgefragten Ports, also der Wert, den das Gerät bei einer Anforderung auf dem Datenbus (oder einzelnen Bits davon) anlegt, wird in das vor der Adresse stehende Register geschrieben.

Bei der Ausgabe an ein Peripheriegerät (z.B. den Lautsprecher) wird der Befehl OUT (Adresse),Wert verwendet. Auch hier kann die Adresse durch einen 8-Bit-Wert oder den 16-Bit-Inhalt des BC-Registerpaares gekennzeichnet werden. Als zu übertragender Wert an das Peripheriegerät kann entweder eine feste Zahl oder ein Register angegeben werden. Dieser Wert wird dann gleichzeitig mit der Adresse (diese liegt auf der Adreßleitung) an den Datenbus ausgegeben und ausgewertet. Auch hier ist zu beachten, daß schon der Zustand eines einzelnen Datenbits das Gerät ansprechen kann (z.B. Bit 4 der Portadresse 254 den Spectrum-Lautsprecher, s. Tabelle). Im Programmteil STEUER werden nun die Tastatur-Ports der Cursor-Tasten abgefragt (Zeilen 780-820), anhand ihrer Bitwerte wird die aktuelle Position der Figur geändert. War zusätzlich noch die Taste 0 gedrückt, so wird dies in der durch RETURN gekennzeichneten Speicherstelle signalisiert, wie auch im Label AENDER eine Änderung der Position dem PRINT-Teil mitgeteilt wird.

Liste einiger IN- und OUT-Adressen

Adr. 254

0. Adreßbit auf null

Bit 0,1,2	BORDER-Farbe setzen
Bit 3	MIC-Eingang abfragbar
Bit 4	Lautsprecher ein/aus
Bit 6	EAR-Ausgang schaltbar

Adr. 253

1. Adreßbit auf null: Ausgabe an den Sinclair-Drucker, Adreßbit 3 und 4 für Microdrive

Adr. 223

5. Adreßbit auf null: Abfrage des Kempston-Joysticks, Bits 0-4 müssen abgefragt werden

Im Sinclair-Handbuch, Kapitel 23 sind die Tastatur-Portadressen angegeben.

Am Ende des Programms stehen dann die ganzen bereitgestellten Speicherstellen, die durch ein Label und die Assemblerbefehle DEFB oder DEFW festgelegt werden. Neben den bisher schon besprochenen Labels sind auch Daten über den Aufbau der Figur (hier stellen sie den Schriftzug "CK" dar) abgelegt, wobei als letztes Byte der Wert FFh steht, damit die PRINT-Routine das Ende der Liste erkennt (Zeilen 610-640). An dieser Stelle sind die Zeilen 400 und 410 zu beachten. In Zeile 400 wird mit dem Label FIGUR die tatsächliche Adresse dieses Labels geladen, während in 410 mit (AKTPOS) der Wert des Bytes bei der Adresse des Labels AKTPOS geladen wird (in diesem Fall sogar die beiden Bytes bei der Adresse).

Das abgedruckte Programm und die Tabelle mit den IN- und OUT-Adressen dürften allen Lesern wieder ein großes Experimentierfeld eröffnen, so daß diese Lektion des Maschinensprachekurses hier beendet werden kann. Im nächsten Teil wird die Serie dann mit der Erklärung der Sonder- und Blockbefehle abgeschlossen.

Thomas Berthold

Figur-Steuerung

```
50 ; Hauptsprungleiste
60 ;
65         CALL PRINT1      ; zeichne Figur
70         XOR  A           ; loesche A
80         LD   (RETURN),A  ; loesche Endekennungsbyte
90 START   CALL BREAK       ; Programm beenden ?
100        CALL PRINT       ; Figur zeichnen
110        CALL STEUER      ; Bewegen der Figur
120        JR   START       ; Schleife wiederholen
130 ;
140 ;
150 ; BREAK-Routine
160 ; Abfrage BREAK-Taste und Label RETURN
170 ;
180 BREAK  CALL #1F54       ; Aufruf ROM-Routine
190        JR   NC,BREAK1   ; Shift + Space ist gedrueckt
200        LD   A,(RETURN)  ; Lade Endekennungsbyte
210        CP   #55         ; Vergleich ob auf Ende gesetzt
220        RET  NZ          ; Kein Ende, Routine verlassen
230 BREAK1 POP  HL          ; vor beenden Stack normalisieren
240        RET  ; Programm beenden
250 ;
260 ;
270 ; PRINT-Routine
280 ; schreibt Figur an aktuelle Position
290 ;
```

```
300 PRINT  LD   A,(AENDER)  ; Lade Aenderungskennung
310        CP   #44         ; Vergleich ob geaendert gesetzt
320        RET  NZ          ; nicht gesetzt, nicht zeichnen
330        CALL PRINT2      ; erst loeschen der Figur
340        LD   HL,(NEWPOS) ; Laden neue Position an
350        LD   (AKTPOS),HL ; aktuelle Position
360        CALL PRINT1      ; zeichnen an der neuen Pos.
370        XOR  A           ; Loesche A
380        LD   (AENDER),A  ; Setze Aenderungskennung zurueck
390        RET  ;           ; beenden dieser Routine
400 PRINT1 LD   DE,FIGUR    ; Laden Figurbytesbeginn
410        LD   BC,(AKTPOS) ; Lade aktuelle Position
420 PRINT2 PUSH BC          ; Rette BC auf Stack
430        PUSH DE          ; Rette DE auf Stack
440        CALL UMWAND      ; ermitteln der Bildschirmadresse
450        POP  DE          ; hole DE vom Stack
460        LD   A,(DE)      ; Lade Figurbyte
470        LD   C,#00       ; Loesche C
480        INC  B           ; erhoehe zum richtigen Zaehlen
490 PRINT3 SRL  A           ; rechtsschieben der Figur
500        RR   C           ; Uebertrag in C schieben
510        DJNZ PRINT3      ; weiter schieben ?
520        XOR  (HL)        ; Lade Bildschirmbyte zur Figur
530        LD   (HL),A      ; Lade Figurbyte in Bildschirm
540        INC  HL          ; naechstes Bildschirmbyte
550        LD   A,C         ; lade A mit Uebertrag
560        XOR  (HL)        ; Lade Bildsch. zum Uebertrag
570        LD   (HL),A      ; Lade Uebertrag in Bildsch.
```

```
580        INC  DE           ; Zeiger auf naechstes Figurbyte
590        POP  BC           ; hole Position vom Stack
600        DEC  B            ; veraendere Y-Position um 1
610        LD   A,(DE)       ; Lade Figurbyte
620        INC  A            ; ist Figurbyte Null ?
630        JR   NZ,PRINT2    ; nein, also weiter
640        RET
650 ;
660 ; UP ermittelt Bildschirmadresse von Position in BC
670 ; HL enthaelt die Adresse
680 ; B enthaelt die Position des Bits (=Anz. Schieben)
690 ;
700 UMWAND CALL #22AA        ; Aufruf ROM-Routine
710        LD   B,A          ; Lade B mit Wert von A
720        RET
730 ;
740 ;
750 ; STEUER-Routine
760 ; Abfrage der Coursortasten und setzen der Position
770 ;
780 STEUER LD   BC,#F7FE     ; Lade Portadresse der Tasten 1-5
790        IN   A,(C)        ; Lade den Portwert in A
800        LD   E,A          ; Lade E mit Wert vom A
810        LD   BC,#EFFE     ; Lade Portadresse der Tasten 6-0
820        IN   A,(C)        ; Lade den Portwert in A
830        LD   BC,(AKTPOS)  ; Lade BC mit akt. Pos.
840        BIT  2,A          ; teste 2. A-Bit ("8")
850        JR   NZ,STEU1     ; nicht gedrueckt, also kein ...
860        INC  C            ; ... erhoehen X-Pos. um 1
870 STEU1  BIT  3,A          ; Abfrage Taste "7"
880        JR   NZ,STEU2
890        INC  B
900 STEU2  BIT  4,A          ; Abfrage Taste "6"
910        JR   NZ,STEU3
920        DEC  B
930 STEU3  BIT  4,E          ; Abfrage Taste "5"
940        JR   NZ,STEU4
950        DEC  C
960 STEU4  BIT  0,A          ; Abfrage Taste "0"
970        JR   NZ,STEU5
980        LD   A,#33
990        LD   (RETURN),A   ; setze Endekennzeichen
1000 STEU5 LD   HL,(AKTPOS)  ; Lade HL mit Akt.Pos
1010       AND  A            ; loesche Uebertragsflag
1020       SBC  HL,BC        ; HL=Akt.Pos. minus neue Pos.
1030       RET  Z            ; wenn gleich Routine beenden
1040       LD   (NEWPOS),BC  ; Lade NEWPOS mit neuer Position
1050       LD   A,#44
1060       LD   (AENDER),A   ; setze Aenderkennzeichen
1070       RET
1080 ;
1090 ;
1100 ; Bereich der Programmvariablen
1110 ;
1120 RETURN DEFB #00         ; Endekennzeichen
1130 AENDER DEFB #00         ; Aenderungskennzeichen
1140 NEWPOS DEFW #0000       ; neue PRINT-Position
1150 AKTPOS DEFW #5030       ; aktuelle PRINT-Position
1160 FIGUR  DEFB %11101001   ; Beginn Figurbytes
1170        DEFB %10000100
1180        DEFB %10001001
1190        DEFB %10001010
1200        DEFB %10000111
1210        DEFB %10001001
1220        DEFB %10001001
1230        DEFB %11101001
1240        DEFB #FF         ; Ende Figurbytes
```

Assemblertips für den Spectrum

Teil 11 Treibersoftware für Epson-kompatible Drucker

Der Spectrum mit Interface 1, Discovery oder Wafadrive erlaubt die Verwendung "richtiger" Drucker mit serieller bzw. Centronics-Schnittstelle. Leider wird diese Möglichkeit von der Software in den Interface-ROMs nicht optimal unterstützt. Wir lernen in dieser Folge ein Utility kennen, das folgende Eigenschaften aufweist:

- verhält sich weitgehend wie ein Textkanal
- frei wählbarer rechter Rand (starre Einstellung auf 80 Zeichen je Zeile fällt weg)
- TAB-Kommando wird ausgeführt
- Blockgrafik wird gedruckt
- Basic-Schlüsselwörter können hervorgehoben und Grafik-Buchstaben unterstrichen werden

In der hier wiedergegebenen Form ist die Routine für Epson-kompatible Drucker speziell für einen DMP 1101 angelegt. Ich hoffe, daß das Utility in dieser Version für möglichst viele Drucker geeignet ist. Gegebenenfalls sind kleine Änderungen erforderlich, beispielsweise zur Ausgabe der Zeichen von Code 127 bis 143 (Copyright-Symbol und Blockgrafik). Da nicht alle Blockgrafikzeichen des Spectrum im Epson-Zeichensatz verfügbar sind, werden manche Character durch Übereinanderdrucken von zwei oder drei Zeichen erzeugt. Die Codierung dieser Zeichenfolge ist in der Liste ab Label GRI ST zu finden. Man kann hier ohne weiteres Änderungen vornehmen.

Die Zeilenbreite ist von 1 bis 254 frei wählbar. Gibt man 0 an, so führt die Routine keine Zeilenlängenkontrolle aus, und die Position des rechten Randes läßt sich durch Übermitteln von Steuercodes an den Drucker festlegen. Die Verwendung von TAB ist in diesem Fall allerdings problematisch.

Das Utility wird über Strom #6 angesprochen (LIST #6 und PRINT #6). Wer die Ausgabe lieber direkt auf Strom #3, den LPRINT-Strom legen möchte, ändert im Programmteil INIT die Adresse 23586 in 23580 (tritt an zwei Stellen auf), in den Programmteilen PRINT und PRNT1 LD A,6 in LD A,3 und streicht das OPEN #3,"t" Kommando im Basic-Programm. Der FORMAT-Befehl ist selbstverständlich nur bei Benutzung eines Interface mit RS 232-Schnittstelle erforderlich und muß an die eingestellte Baudrate angepaßt werden.

Das Maschinenprogramm ähnelt in Ablauf und Aufgabenstellung dem 32+40 Print-Programm aus den Assemblertips in CK 8/9 86: Einbinden einer Zeichenausgaberoutine in das Strom- und Kanalsystem des Spectrum. Sieht man sich das Assemblerlisting näher an, entdeckt man – einmal abgesehen von der eigentlichen Character-Ausgabe, die bei Grafikbildschirm und Drucker natürlich grundverschieden ist – interessante Unterschiede. Was hat es mit den EXX-Statements auf sich, und weshalb werden über die benutzten Register zwischenzeitlich auf den Stack gerettet, während das bei der Bildschirmausgabe offensichtlich nicht notwendig war? Zur Beantwortung dieser Fragen führen wir uns noch einmal vor Augen, wie die Zeichenausgabe im Spectrum-Betriebssystem vonstatten geht. Der auszugebende Character (das kann ein ASCII-Zeichen, ein Steuercode, ein Basic-Token oder ein Grafikzeichen sein) wird in den Akku gebracht und durch RST $10 über den gerade geöffneten Strom an die Ausgaberoutine eines Kanals weitergeleitet. Der RST $10-Mechanismus schaltet durch EXX auf das zweite Registerset der Z80-CPU um, ehe die Ausgaberoutine aufgerufen wird, und anschließend auf das erste Registerset zurück. Diese Methode stellt sicher, daß der Inhalt der Register B, C, D, E, H, L im ersten Registersatz durch die Zeichenausgabe nicht verändert wird und sonst nach Ausführung von RST $10 weiterhin zur Verfügung steht.

So weit, so gut. Nun ruft unsere Routine jedoch ihrerseits durch RST $10 eine Ausgaberoutine auf, nämlich die von Strom #4. Ohne weitere Maßnahmen hätte das zur Folge, daß das neuerliche EXX auf den ersten Registersatz zurückschaltet und die Strom-#4-Zeichenausgabe dessen Inhalt zerstören könnte. Daher wechselt unsere Routine von sich aus auf das erste Registerset und rettet seinen Inhalt auf den Stack, bevor weitere RST-$10-Instruktionen folgen. Vor der Rückkehr ins Spectrum-Betriebssystem muß die alte Registersituation wiederhergestellt werden. Außerdem ist es zwingend notwendig, wieder Strom #6 zu öffnen, d. h. zum aktuellen I/O-Strom zu machen, damit auch die noch folgenden Character über unsere Routine und nicht etwa über Strom #4 ausgegeben werden.

Diesmal wird zur Abwechslung ein etwas anderes Basic-Programm zur Erzeugung des Maschinencodes verwendet als in den bisherigen Folgen. Jede einzelne DATA-Zeile enthält als letzte Byte-Information eine Prüfsumme über ihren eigenen Inhalt. Tippfehler lassen sich so genauer lokalisieren und schneller korrigieren.

Bei dieser Gelegenheit möchte ich noch ein paar Worte zu den in dieser Serie immer wieder auftauchenden Basic-Programmen zum Generieren der Maschinenprogramme sagen. Im Laufe der Zeit erhielten wir Briefe und Anrufe von CK-Lesern, die Probleme mit diesen Programmen hatten. Ihre Schwierigkeiten ließen sich immer auf Tippfehler zurückführen. Man muß sich darüber im klaren sein, daß ein Eingabefehler nicht unbedingt zu der Meldung „Fehler in A$" führt. Je nach Art der Falscheingabe kann das Basic-Programm z. B. auch mit der Fehlermeldung "Number too big" oder "Integer out of range" abbrechen. Wichtig ist auch die Kleinschreibung in den LET A$- bzw. DATA-Zeilen. Wer unbedingt Großbuchstaben eintippen möchte, muß in der DEF FN A-Anweisung die 39 durch eine 7 ersetzen (Großbuchstaben beginnen mit Code 65 = 48 + 10 + 7, Kleinbuchstaben mit 97 = 48 + 10 + 39).

Michael Schramm

Epson-Druckertreiber

```
  1 REM Treibersoftware fuer
        Epson-Drucker am
        Spectrum

        Michael Schramm
        Computer Kontakt 1986

  10 RESTORE : CLEAR 64939
  20 DEF FN A(X$)=CODE X$-48-39*
(X$>"9")
  25 DEF FN B(X$)=16*FN A(X$(1))
+FN A(X$(2))
  30 PRINT "Treibersoftware-Masc
hinencode   wird erzeugt - dauer
t nur einen Augenblick."
  40 LET A=64940
  50 FOR Z=1000 TO 1030 STEP 10
  60 PRINT "Zeile ";Z;": ";
  70 READ A$: LET S=0
  80 IF LEN A$<4 OR LEN X$/2<>IN
T (LEN A$/2) THEN GO TO 200
  90 FOR I=1 TO LEN A$-2 STEP 2
 100 LET X=FN B(A$(I TO I+1))
 110 LET S=S+X
 120 POKE A,X: LET A=A+1
 130 NEXT I
 140 IF FN B(A$(I TO ))<>S-256*I
NT (S/256) THEN PRINT "fehlerhaf
t!": STOP
 150 PRINT "ok"
 160 NEXT Z
 170 PRINT "Maschinencode ist in
 Ordnung."
 180 INPUT "Zeilenbreite: ";Z
 190 INPUT "Basic-Schluesselwoer
ter fett-   drucken (J/N)? ";A$
 200 LET F=(A$="J" OR A$="j")
 210 INPUT "Graphik-Buchstaben u
nterstrei-  chen (J/N)? ";A$
 220 LET U=(A$="J" OR A$="j")
 230 CLOSE #3: CLOSE #4: CLOSE #
6: OPEN #3;"t": OPEN #4;"b"
 240 FORMAT "b";4800
 250 PRINT USR 64940,Z+512*(F+2*
U)
 260 PRINT
 270 PRINT "Ausgabe ueber Strom
#6"
 280 STOP
1000 DATA "cd811ccda22dc52a225c7
cb52802cf172e535ce52b010b00e5cd5
516e11103fe7323722336c4233615233
642230110b00712370232323712370e1e
d5b4f5ca7ed5222225c412100fe73237
02372ed7b3d5cc3761b20000f5cd541
fd27b1b3e04cd0116d9e360"
1010 DATA "d5c554c462fe3e06cd011
6c1d1e1d9c92b7ec610e6f01805cb86c
bfe2b2b4ecb014e232806b938039118f
a574604b8dc59fe7a962806473e20471
0fb72c92b7ea7cd353e08d7c92b36003
e0dd73e0ad7c92102fe7ea7cbbef81f7
a38bbfe06d828aefe084897"
1020 DATA "28d7fe04d828dbfe10d8c
bfefe16d8cbbecbc6fe18d0cb86fe20d
8d6a53823cb4ef528063e1bd73e45d73
```

```
e06cd0116f1f5cd100cf1c83e04cd011
63e1bd73e46d7c92b342b7e23a728069
635dc59fe347a23d67f300f7afdcb018
6fe202804fdcb01c6d7c932"
1030 DATA "fe1130192115ff47047e2
3a720fb10f918067ea7c83e00d77e23d
718f5cb56f528093e1bd73e2dd73e01d
77ad64fd7f1c83e1bd73e2dd7afd7c90
0fe0020000f100f000f0f100f300f1f30
0f0f300f0f1f300f200f400ab00abf10
0a300a3f100abf300a700c7"
9000 SAVE "Druck.BAS" LINE 1
9010 STOP
9500 SAVE "Druck.CODE" CODE 6494
0,404
9510 STOP
```

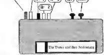
Assembler-Listing

```
        ORG 64940

; 8000 Treibersoftware für Epson Drucker

ERRSP  = 23613
CHANS  = 23631
PROG   = 23635

MKROOM   = #1655
BRKST    = #1F54
REP_L    = #1B7B
OPEN     = #1601
TOKPR    = #0C10
STACKIN  = #1E99
UNSTACK  = #2BF1
STMRET   = #1B76

BS  =  8
LF  = 10    ;Drucker
CR  = 13    ;Steuercodes
ESC = 27

INIT  CALL STACKIN   ;Param -> 1 Stapel
      CALL UNSTACK   ;Parameter in BC
      PUSH BC        ;Auf Stack retten
      LD HL,(23582)  ;Falls Strom #4
      LD A,H         ;schon benutzt ist,
      OR L           ;Fehlermeldung
      JR Z,INIT1
      RST #08
      DEFB #17
INIT1 LD HL,(PROG)   ;Bereich im CHANS-
      PUSH HL        ;Bereich 11 Bytes
      DEC HL         ;einfügen und eine
      LD BC,11       ;Kanalbeschreibung
      PUSH HL        ;aufbauen (siehe
      CALL MKROOM    ; auch Assemblertips
      POP HL         ; in CK 8-9/1986)
      LD DE,PRINT
      LD (HL),E      ;Die #4-Ausgabe wird
      INC HL         ;über die Routine
      LD (HL),D      ;PRINT abgewickelt
      INC HL         ;werden.
      LD (HL),#C4
      INC HL
      LD (HL),#15
      INC HL
      LD (HL),"P"
      INC HL
      LD BC,11
      LD (HL),C
      INC HL
      LD (HL),B
      INC HL
      INC HL
      INC HL
      LD (HL),C
      INC HL
      LD (HL),B
      POP HL
      LD DE,(CHANS)
      AND A
      SBC HL,DE
      LD (23582),HL
      POP DE          ;Die vom Basic
      LD HL,RAND      ;übergebenen Parame-
      LD (HL),E       ;ter in RAND und
      INC HL          ;PFLAG eintragen
      LD (HL),0
      INC HL
      LD (HL),D
      LD SP,(ERRSP)
      JP STMRET

RAND  NOP            ;Pos. rechter Rand
XPOS  NOP            ;Akt. Druckposition
PFLAG NOP            ;Flags
      ;Die Bedeutung der Bits von PFLAG:
      ; 0: TAB Position folgt
      ; 1: Basic-Token fettdrucken
      ; 2: Graphikzeichen unterstreichen
      ; 7: Nächsten Character ignorieren

PRINT PUSH AF        ;Character retten
      CALL BRKST     ;Abbruch, falls
      JP NC,REP_L    ;BREAK gedrückt
      LD A,4         ;Zeichenausgabe über
      CALL OPEN      ;Strom #4
      EXX            ;Hauptregistersatz
      EX (SP),HL     ;einschalten und
      PUSH DE        ;alle Register auf
      PUSH BC        ;den Stack retten
      LD E,H         ;Character in E
      CALL PRNT1     ;Zeichenausgabe
      LD A,2         ;Wieder auf Strom #2
      CALL OPEN      ;umschalten und die
      POP BC         ;alte Registerinhalte
      POP DE         ;wieder her-
      POP HL         ;stellen, bevor der
      EXX            ;Rücksprung erfolgt
      RET

PRCOM INC HL         ;Behandlung des Kom-
      LD A,(HL)      ;mas als Separator
      ADD A,16       ;beim Drucken (16er
      AND #F0        ;Schritt Tabulator)
      JR TABU1

TABUL RES 7,(HL)     ;Nächsten Character
      SET 0,(HL)     ;ignorieren
      DEC HL         ;HL auf RAND bei
TABU1 INC HL         ;(RAND)=0 wird die
      LD C,(HL)      ;TAB Position direkt
      RLC C          ;genommen
      LD C,(HL)
      INC HL
      JR Z,TABOK
TABCP CP C           ;Setzt TAB Position
      JR C,TABOK     ;modulo (RAND)
      SUB C
      JR TABCP

TABOK LD E,A         ;Endgültige TAB Pos
      LD B,(HL)      ;in E
      INC B          ;Ausgabestelle
      CP B           ;Zeilenvorschub (ZV)
      CALL C,ENTR1   ;ausführen
      LD A,E         ;Falls ZV ausgef. &
      SUB (HL)       ;TAB Position = 0,
      JR Z,TABEN     ;schon fertig
      LD B,A         ;Sonst die erforder-
TABLP LD A," "       ;liche Anzahl von
      RST #10        ;Blanks drucken, um
      DJNZ TABLP     ;die Pos. anzust.
TABEN LD (HL),E      ;Akt. Pos. des Dr-
      RET            ;kopfes in XPOS

PRBS  DEC HL         ;HL auf XPOS
      LD A,(HL)      ;Backspace nur aus-
      AND A          ;führen, sofern die
      RET Z          ;Printposition
      DEC (HL)       ;größer als null
      LD A,BS        ;BS
      RST #10
      RET

ENTER INC HL         ;HL auf XPOS
ENTR1 LD (HL),0      ;CR und LF ausgeben
      LD A,CR        ;um einen Zeilenvor-
      RST #10        ;schub zu bewirken
      LD A,LF
      RST #10
      RET

PRNT1 LD HL,PFLAG    ;Zunächst wird PFLAG
      LD A,(HL)      ;ausgewertet. Bit 7
      AND A          ;rücksetzen. Char
      RES 7,(HL)     ;ignorieren, falls es
      RET M          ;vorher gesetzt war
      RRA            ;Falls Bit 0 gesetzt,
      LD A,D         ;Char als TAB-Pos
      JR C,TABUL     ;interpretieren
      CP 6           ;Char von 0 bis 5
      RET C          ;ignorieren
      JR Z,PRCOM     ;6 = Printseparator,
      CP 8           ;7 ignorieren
      RET C
      JR Z,PRBS      ;8 = Backspace
      CP 13          ;9 bis 12 ignorieren
      RET C
      JR Z,ENTER     ;13 neue Zeile
      CP 16          ;14 und 15 ignorier
      RET C          ;16 bis 21 mitsamt
      SET 7,(HL)     ;Folgecharacter
      CP 22          ;ignorieren
      RET C
      RES 7,(HL)     ;23 = TAB
      SET 0,(HL)     ;22=AT sollte nicht
      CP 24          ;verwendet werden
      RET C
      RES 0,(HL)     ;24 bis 31 wird
      CP " "         ;ignoriert
      RET C
      SUB 165        ;Falls kein Basic-
      JR C,GRAPH     ;Token vorliegt,
      BIT 1,(HL)     ;bei GRAPH weiter
      PUSH AF        ;Falls Bit 1 in
      JR Z,NBOLD     ;PFLAG gesetzt ist,
      LD A,ESC       ;werden die Token-
      RST #10        ;Character fettge-
      LD A,"E"       ;druckt
      RST #10        ;Die einzelnen Char.
NBOLD LD A,4         ;über Strom #4 ohne
      CALL OPEN      ;diese Routine,
      POP AF         ;ausgeben
      PUSH AF        ;Token Expansion
      CALL TOKPR     ;durch ROM-Routine
      POP AF         ;Sofern kein Fett-
      RET Z          ;druck, fertig
      LD A,4         ;Andernfalls wieder
      CALL OPEN      ;auf Normaldruck
      LD A,ESC       ;umschalten
      RST #10
      LD A,"F"
      RST #10
      RET

GRAPH DEC HL
      INC (HL)       ;XPOS erhöhen
      DEC HL
      LD A,(HL)      ;(RAND) -> Akku
      INC HL         ;HL auf XPOS
      AND A          ;Falls (RAND) = 0,
        JR Z,GRPH3     ;sein Zeilenvorsch
        SUB (HL)       ;Sonst rechten Rand
        DEC (HL)       ;berücksichtigen
        CALL C,ENTR1
        INC (HL)
GRPH3   LD A,E         ;Character -> Akku
        INC HL         ;HL auf PFLAG
        SUB 127
        JR NC,BLKGR    ;Normale ASCII-Char
        LD A,E         ;(32 bis 126) werden
        RES 0,(IY+1)   ;hier ausgegeben
        CP " "         ;Bit 0 in FLAGS ist
        JR NZ,NBLNK    ;bei Vorliegen eines
        SET 0,(IY+1)   ;Blanks zu setzen,
NBLNK   RST 010        ;sonst rückzusetzen
        RET

BLKGR   CP 17          ;Hier die Zeichen
        JR NC,NBLGR    ;127 bis 143 behand
        LD HL,NBLST
        LD B,A         ;HL wird auf die zum
        INC B          ;Graphikzeichen ge
GRCH    LD A,(HL)      ;hörende Char.-folge
        INC HL         ;in der Liste NBLST
        DEC A          ;gesetzt.
        JR NZ,GRCH     ;Den ersten Char
        DJNZ GRCH      ;der Folge sofort

        JR NOBS        ;drucken, die weite
                       ;ren mit BS davor
GRLOP   LD A,(HL)      ;Character in den
        AND A          ;Akku
        RET Z          ;bei Null fertig.
        LD A,BS        ;Die Character der
        RST 010        ;Reihe nach überein
NOBS    LD A,(HL)      ;anderdrucken
        INC HL
        RST 010
        JR GRLOP

NBLGR   BIT 2,(HL)     ;Die Ausgabe der
        PUSH AF        ;Graphikbuchstaben
        JR Z,NOUND     ;Falls Bit 2 in
        LD A,ESC       ;PFLAG gesetzt ist,
        RST 010        ;Steuercode für
        LD A,"-"       ;Unterstreichung
        RST 010        ;ausgeben
        LD A,1
        RST 010
NOUND   LD A,E         ;Den Graphikbuchsta
        SUB 79         ;ben in einen norma-
        RST 010        ;len Buchstaben
        POP AF         ;verwandeln und
        RET Z          ;drucken
        LD A,ESC

        RST 010        ;Erforderlichenfalls
        LD A,"-"       ;die automatische
        RST 010        ;Unterstreichung
        XOR A          ;wieder ausschalten
        RST 010
        RET

NBLST   DEFB 0         ;Liste der Charact.,
        DEFB 0FF,0     ;die statt der Zei
        DEFB 0EF,0     ;chen mit Codes von
        DEFB 0F1,0     ;127 bis 143 ausge-
        DEFB 0F0,0     ;geben werden
        DEFB 0F0,0F1,0
        DEFB 0F1,0
        DEFB 0F1,0F1,0
        DEFB 0F3,0F2,0
        DEFB 0F0,0F1,0F2,0
        DEFB 0F2,0
        DEFB 0F0,0     ;Jede Charact.folge
        DEFB 0E2,0     ;für einen Zeichen
        DEFB 0E2,0F1,0 ;wird durch ein
        DEFB 0A0,0     ;Nullbyte abge
        DEFB 0A3,0F1,0 ;schlossen
        DEFB 0A8,0F3,0
        DEFB 0A7,0

        END
```

ISO-FACE 1,0 – die neue EPROM-Steckkarte für den Spectrum

In der Januar Ausgabe der CK hatten wir das ISO ROM von Individual Software vorgestellt. Aus dem gleichen Haus gibt es jetzt das ISO-FACE 1,0. Dieses Gerät bietet dem Anwender die Möglichkeit bis zu drei zusätzliche ROMs einzusetzen, ohne einen Eingriff im Spectrum vornehmen. Es handelt sich also um eine EPROM Steckkarte. Optisch präsentiert sich der Zusatz eher unauffällig. Eine Black Box in der Art eines Joystick- oder Druckerinterfaces wird als erste Erweiterung an den Spectrum gesteckt. Die Anschlußleiste ist natürlich durchgeführt, so daß weitere Peripheriegeräte angeschlossen werden können. Oben befinden sich einige Schalter:

- der Umschalter für internes/externes ROM
- ein Schiebeschalter für die Wahl eines der drei möglichen EPROMs
- ein NMI (nichtmaskierbarer Interrupt für Warmstart)-Taster
- ein Reset Taster (für Kaltstart)



Die Taster und ihre Bedeutung

Der Spectrum Benutzer hat jetzt erstmals die Gelegenheit sein Original ROM zu benutzen, einen Eingriff in den Computer zu vermeiden (ein evtl. Garantieverlust wird verhindert) und trotzdem mit anderen ROMs zu arbeiten. Dabei spielt es keine Rolle, ob eigene EPROMS erstellt werden oder z.B. die ISO ROMs zum Einsatz kommen. Verwendet werden können die EPROM Typen 2716 bis 27128. Durch die Installation prellfreier Taster können die Möglichkeiten der ISO ROMs (Monitor oder Back Up-Routine) direkt genutzt werden, zusätzliche Lötarbeiten am Spectrum entfallen. Unter dem Original ROM steht dann auch endlich ein RESET-Taster wie beim Spectrum+ zur Verfügung. Mit einem voll bestückten ISO-Face hat man die Wahl zwischen vier ROMs, was für die meisten Anwender ausreichen dürfte. Das Umschalten zwischen internem und externem ROM erfolgt absturzfrei, lediglich die Wahl eines EPROMs muß in der Stellung "internes ROM" erfolgen.

Einfaches aber schönes Innenleben

Zum Test waren dem ISO Face das Monitor- und das Back Up-ROM mitgegeben. Die Umschaltung und die Arbeit mit dem Original ROM und externen EPROMs funktionierte einwandfrei. Bei dieser Gelegenheit weist der Hersteller darauf hin, daß die ISO ROMs verbessert wurden. Das Back Up-ROM ist jetzt speziell auch auf Beta Disk Benutzer zugeschnitten. Außerdem wurde eine Routine implementiert, die den gesamten RAM-Speicherinhalt komprimiert. Das Monitor ROM wurde ebenfalls verbessert. Das Adressenformat für den Monitor ist jetzt HI LO (wie bei der Eingabe von Hexadezimalzahlen in Assembler). Außerdem kann beim Ausdrucken eine beliebige Zeichenbreite gewählt werden.

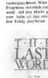
Wer mit seinem eigenen ROM nicht mehr zufrieden ist und einen Eingriff in seinen Spectrum bisher gescheut hat, ist mit dem ISO Face (ebenso mit den ISO ROMs) gut bedient. Der Preis von ca. 80,- DM erscheint angemessen.

Weitere Informationen:
Individual Software V. Mirolin
Schäferstr. 41
4600 Dortmund 30

Rolf Knorre

Eines der besten Textverarbeitungsprogramme für den Spectrum

Es gibt mittlerweile eine ganze Reihe von Textverarbeitungsprogrammen für den ZX Spectrum. Das bekannteste und beliebteste ist wohl "Tasword Two", das wohl fast jeder Spectrum-Benutzer in seiner Bibliothek hat. Die englische Firma SAGA Systems hat sich dadurch aber nicht abschrecken lassen und vor kurzer Zeit ein neues Programm auf den Markt gebracht, das die Bezeichnung "The Last Word" trägt. Um es vorwegzunehmen: Wäre dieses Programm zeitgleich mit "Tasword" auf den Markt gekommen, hätte ich ihm einen großen Erfolg prophezeit. Heute bin ich mir nicht mehr sicher, ob es viele User gibt, die von einem bewährten Produkt so einfach umsteigen wollen. Trotzdem: "The Last Word" ist hervorragend.



Das Programm besteht in der Hauptsache aus einem rund 14,5 KByte langen Maschinencodeteil, der von Cassette geladen wird. Als Textspeicher verbleiben 22000 Byte. Neben dem Hauptprogramm befinden sich auf der Cassette noch einige Files, die der Anpassung an verschiedene Massenspeicher und Drucker-Interfaces dienen. Berücksichtigt wurden folgende Systeme:

- Cassettenrecorder/Microdrive (bereits in "TLW" enthalten)
- Kempston KDOS
- Beta Disc
- Opus Discovery
- Gordon (mir leider unbekannt)
- Wafadrive
- Kempston S Interface
- Sinclair Interface I RS 232 Port

Ich habe das Programm mit einem Epson-kompatiblen Drucker und dem Kempston-E-Interface getestet. Für diese Zusammenstellung werden die o.g. Routinen nicht benötigt, man kann also nach Laden des Hauptprogramms direkt arbeiten.

Nun zu den Möglichkeiten, die "The Last Word" dem Anwender bietet. Vorab sei noch gesagt, daß ich unmöglich alle Punkte ausführlich beschreiben kann, da sonst der Rahmen dieses Tests gesprengt würde. Ich hoffe aber, daß Interessenten genügend Informationen erhalten.

Beginnen möchte ich mit dem aktuellen Arbeits-Screen, der immer auf dem Bildschirm bleibt. Die unteren 20 Zeilen sind für den Text reserviert. Am oberen Rand befinden sich drei Statuszeilen, die permanent folgende Daten anzeigen:

Links: Zeilennummer / Spaltennummer / freier Speicherplatz

Mitte: Gewählte Zeichen pro Zeile / Groß- und Kleinschreibung / linker und rechter Rand / Tabulatorstop / Modus. In diesem Bereich werden auch besondere Bearbeitungshinweise gegeben.

Rechts: Anzeige für Word Wrap an bzw. aus / Blocksatz an bzw. aus / Überschreiben / Einfügen

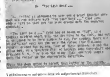
Die Programmierer haben versucht, ihrem Produkt größtmöglichen Komfort und leichte Bedienbarkeit mitzugeben. Aus diesem Grund wurde auch darauf verzichtet, mit Menüs zu arbeiten; alle Funktionen werden über Tastenkombinationen aufgerufen. Notwendige Einblendungen und Hinweise erfolgen im oberen Statusteil. Der Anwender kann also ständig seinen Text im Auge behalten. Eine Ausnahme bildet hier eine Helppage, die auf Wunsch über den Text gescrollt werden kann.

Wer sich ein wenig mit Textverarbeitungsprogrammen auskennt, wird die schriftliche Anleitung (leider zur Zeit nur in Englisch erhältlich) kaum benötigen, aber auch ein totaler Neuling kann sehr schnell die ersten Texte erstellen. Die Zeichendarstellung auf dem Bildschirm kann der vorhandenen Hardware angepaßt werden. Auf Tastendruck stehen dem Anwender 40, 60 oder 80 Zeichen zur Verfügung. 80 Zeichen waren aber selbst auf einem Monitor nicht gerade angenehm für die Augen. Optimal ist die 60-Zeichen-Variante, die sich auch als Briefformat eignet. Auch die BORDER-, PAPER- und INK-Farben können vom Programm aus geändert werden.

Nun zu den Bearbeitungsbefehlen, die direkt über die Tastatur aufgerufen werden (in der Regel mit der Symbol-Shift-Taste):

- Zeichen nach rechts/links
- Zeile nach oben/unten (über Cursor-Tasten)
- Tabulatorschritte
- Wort oder Zeile nach links/rechts
- Sprung an Textanfang oder -ende
- Sprung zu einer einzugebenden Zeilennummer
- Textscroll nach oben/unten
- Löschen eines Zeichens, Wortes, Textes oder einer Zeile
- Umschaltung Text einfügen/überschreiben
- Text nach links oder rechts schieben
- Text zentrieren und einiges mehr

Damit sind alle Optionen, die man von anderen Programmen her kennt, vorhanden. Auch weitere Standardfunktionen, wie z.B. das Einstellen der Ränder oder Setzen von Tabulatorstops, sind natürlich möglich. Das Word Wrapping, also der Umbruch eines Wortes, das den Rand überschreiben würde, funktioniert tadellos, ebenso der dabei ausgeführte Zeilenausgleich (=Blocksatz: die in einer Zeile vorhandenen Wörter werden auf die ganze Zeilenbreite verteilt). Beide Funktionen sind abschaltbar. Steuercodes, die dem Drucker z.B. eine andere Schriftart mitteilen, werden direkt in den Text geschrieben. Die dabei auftauchenden Grafiksymbole können ebenso wie die Zeilen



Viel Information auf einem dennoch aufgeräumten Bildschirm und und Word Wrap-Marker ausgeblendet werden, so daß auf dem Bildschirm nur noch der reine Text verbleibt.

Jetzt zu einigen Besonderheiten, die andere Programme dieser Art nicht bieten. Mit dem Befehl WORD COUNT kann z. B. abgefragt werden, wie viele Wörter und Zeichen man bereits geschrieben hat und in welcher Position der Cursor momentan steht. Interessanter ist aber CALCULATOR. Damit steht ein kompletter Taschenrechner zur Verfügung, der allerdings nur im oberen Statusteil angezeigt wird. Neben den Grundrechenarten kann eine Vielzahl weiterer Operationen durchgeführt werden. Dazu dienen die Befehle SIN, COS, TAN, ASN, ACS, ATN, EXP, SQR, INT, LN, VAL, SGN, ABS, PI, >, <, <>, >=, OR, AND und NOT. Das Ergebnis einer Rechnung kann nun entweder gelöscht, in einen der neun Speicher übernommen oder direkt in den Text eingesetzt werden. Die Zahl erscheint immer an der aktuellen Cursorposition. Wer Gefahr läuft, zu lange zu arbeiten, kann die interne Uhr abrufen oder die Zeit für einen akustischen Alarm einstellen.

Die Printer Option ist ebenfalls vorbildlich. Was gedruckt werden soll, wird über die Zeilennummern eingegeben. Wird eine Kopie benötigt, kann dies auch vermerkt werden (bis zu 254 Kopien sind möglich); auch der Zeilenabstand kann eingestellt werden. Ähnlich arbeitet auch die SAVE/LOAD-Option. Neben der Wahl zwischen Microdrive und Cassette (oder dem eigenen Speichermedium) kann man z. B. angeben, welche Zeilennummern abgespeichert werden sollen.

"The Last Word" gehört mit zu den besten Textverarbeitungssystemen für den Spectrum. Die einfache Handhabung und die geringe Anfälligkeit gegen Bedienungsfehler, die große Leistungsfähigkeit und der hohe Komfort haben mich voll überzeugt. Warum ist dieses Programm nicht früher auf den Markt gekommen?

System Spectrum 48 K
Hersteller SAGA Systems

Rolf Knorre

Graf Eddi – ein Grafikeditor besonderer Art

Um eines von vornherein klarzustellen: "Graf Eddi" ist kein Programm zum Malen von Bildern. Man kann damit aber vorhandene Grafiken bearbeiten, so z. B. spiegeln, drehen, Teile löschen, invertieren, heraustrennen, verschieben, kopieren, Texte einblenden, Farben ändern, verschiedene Grafiken übereinanderlegen und das Bild den Farben gemäß schattiert auf einem Sinclair-kompatiblen Drucker als 24-Zeilen-Copy oder auf einem Normalpapierdrucker ausgeben.

"Graf Eddi" verwaltet Bildschirmspeicher, Workingpage, Sicherheitskopie und zwei Alternativseiten. Ferner stehen 2 Zeichensatzseiten bereit. Farbinformationen werden getrennt vom übrigen Bild verwaltet, so daß die Möglichkeit besteht, die Farben des Bildes durch ein Schachbrettmuster zu ersetzen, um dann später die Originalfarben wieder zurückzuholen. Alle wesentlichen Teile des Programms sind in Maschinencode geschrieben. Geschwindigkeitsprobleme entstehen somit nicht.

"Graf Eddi" besteht aus 2 Listings. Listing 1 ist ein Hilfsprogramm, um den ca. 1500 Byte langen Maschinencodeteil des Programms zu laden. Speichern Sie es nach dem Eintippen mit GOTO 9000 ab. Danach löschen Sie den Speicher und geben Listing 2, den Basic-Teil, ein. Speichern Sie diesen mit GOTO 9880 ab. Nun laden Sie wiederum den MC-Lader und starten ihn mit RUN. Fehler in den DATA-Zeilen werden erkannt. Treten keine auf, fordert Sie das Ladeprogramm nach einiger Zeit dazu auf, Programm 2 zu laden. Kommen Sie dieser Aufforderung nach und speichern Sie das gesamte lauffähige Programm mit GOTO 9820 ab.

Die einzelnen Funktionen von "Graf Eddi" werden angewählt, indem man den Cursor auf den Schriftzug plaziert und Enter oder 0 drückt. Der kreuzförmige Cursor, der bei Pixel-Operationen eingeblendet wird, kann ebenfalls mit den Cursor-Tasten und 0 pixelweise bewegt werden (in Verbindung mit Symbol Shift um 5 Pixel). Mit der Taste 1 kommt man von diesem Cursor wieder ins Menü. Zu bearbeitende Bildausschnitte werden – so nicht anders erwähnt – durch die linke untere und die rechte obere Ecke mit Hilfe des kreuzförmigen Cursors und der Taste 0 markiert. Grundsätzlich können sich Ursprungs- und Zielbildschirmteil ohne Beeinträchtigung überlagern.

Die Funktionen haben folgende Bedeutungen:

CHAR 1: Alternativzeichensatz 1

CHAR 2: Alternativzeichensatz 2

NORMCHAR: Sinclair-Zeichensatz

ROTATE: Der quadratische Bildschirmteil, markiert durch die linke untere und linke obere Ecke, wird gedreht.

MIRROR H: Der markierte Bildschirmteil wird an die Stelle, deren rechte untere Ecke festgelegt wurde, horizontal gespiegelt.

MIRROR V: Wie MIRROR H, es wird aber vertikal gespiegelt.

COPY: Der markierte Bildschirminhalt wird an die Stelle kopiert, deren linke untere Ecke festgelegt wurde.

MOVE: Wie COPY, der Ursprungsbildteil wird aber gelöscht.

Raparatur-Schnelldienst/Ersatzteile für Spectrum! Rufen Sie uns an!

Spectrum-Eprommer

248.–

- ★ 2716/2732/2732A/2764/27128
- ★ Intelligente Programmierung
- ★ Menü gesteuert
- ★ zus. Centronics-Schnittstelle
- ★ Keine zus. Hardware erf.
- ★ Preis 248.– DM inkl. Software für EPROMMER und Drucker
- ★ Spez. Sockel zum Austausch Spectrum-ROM – 27128 **24.80 DM**
- ★ EPROM-Löschgerät **117.00 DM**
- ★ Druckerkabel Centronics **37.00 DM**

Alle Preise zuzügl. Versandkosten. Info anfordern.

ZX-Spectrum
C&M Meyer
Rahserstraße 50
4060 Viersen 1
☎ 0 21 62 / 2 29 64

Jupitersoft GdbR

ZX Spectrum - ZX81
C 16-Software
ab **7.90 DM**

Adventures, Action, Anwender, Grafik und Dateiprogramme – C 16 auch auf Disk

Viel Software stark im Preis reduziert.

Kostenloser Katalog bei

Bernd Denk
Jupitersoft GdbR
Frühlingstr. 12
8831 Weiboldshausen
☎ 09141/2249 ab 16 Uhr

unicorn soft
OHG Rudolph
Versand:
Ehlenerstr. 7
3501 HOOF
Verkauf:
Druseltore
3500 KASSEL
0561/770367

ATARI ST Tagestiefstpreis
Spectrum ab 248.-
Sinclair QL ab 448.-
Tastatur f. QL ab 198.-
Opus Discovery ab 589.-
Seikosha SP 1000 ab 698.-
Computer, Drucker + Zubehör in unserer Gratisinfo
Unseren Kunden
ein frohes Weihnachtsfest
Ursula & Joachim Rudolph

DELETE Der markierte Bildschirmteil wird gelöscht

FETCH Der markierte Bildschirmteil wird in die Alternativseite eingeblendet (ohne Farben)

INVERSE Der markierte Bildschirmteil wird invertiert

ZOOM Ein durch die linke obere Ecke markierter 32*22 Pixel großer Bildschirmteil wird 8mal vergrößert und kann bearbeitet werden. Der Cursor hat dabei eine OVER Funktion. Zurückgeblendet wird in Originalgröße in das Gesamtbild, indem man den Cursor auf den Schriftzug STORE bringt. Ohne Veränderung erreicht man das Menü durch EXIT

ATTR Ruft folgende Untermenüpunkte auf

1 ATTRIBUT Blendet den externen Attributspeicher in den Farbspeicher der Workingpage
2 CHESS Blendet ein Schachbrettmuster ein
3 SET ATTR Nach Eingabe der Farbwerte können mit dem Cursor Attribute verändert werden
4 EXIT Bringt das Hauptmenü zurück

RECALL Holt die Sicherheitskopie in die Workingpage

BACKUP Kopiert die Workingpage in die Sicherheitskopie

LD 1--S Kopiert die Workingpage in die Alternativseite 1

PRINT Fügt Texte in das Bild der Workingpage ein. Dabei kann es in den untersten 2 Zeilen zu Farbverfälschungen kommen, die aber durch Zurückblenden der Farben mittels ATTR/ATTRIBUT wieder rückgängig gemacht werden können

Mit CAPS SHIFT und Taste 9 schaltet man auf den Grafikmodus um. In dem deutsche Sonderzeichen (Tasten A-G) verfügbar sind

EXCHAB Vertauscht Alternativzeichensätze miteinander

ORS--1 Blendet Alternativseite 1 in die Workingpage ein

ORS--2 Blendet Alternativseite 2 in die Workingpage ein

SCHATT Das Bild in der Workingpage wird den Farben des ext. Attributspeichers entsprechend für die Druckerausgabe schattiert

EXS--1 Vertauscht Workingpage und Alternativseite 1

EXS--2 Vertauscht Workingpage und Alternativseite 2

SINCLAIR Gibt eine 24-Zeilen Kopie (!) auf einem Sinclair kompatiblen Drucker aus

EIGEN Ruft eine einsetzbare Hardkopy Routine auf. Der Einsprungbefehl steht dann in Zeile 3500. Die Routine kann im Speicher bei 65090 liegen und darf 300 Byte lang sein

SAVE Speichert nach Eingabe des Namens (nur Enter führt zum Menü) die Workingpage ab

MSAVE Speichert entsprechend SAVE auf Microdrive ab

CAT Microdrive Katalog

LOAD Lädt nach Eingabe des Namens ein Bild, die Länge muß 6912 Byte (Screen$) betragen

MLOAD Lädt ein Bild entsprechend LOAD von Microdrive

ERASE Löscht nach Eingabe des Namens einen Code dieses Namens auf Microdrive

LOAD CH Lädt nach Eingabe des Namens einen Zeichensatz, die Länge muß 768 Byte betragen

MLOAD CH Lädt einen Zeichensatz entsprechend LOAD CH von Microdrive. Bei den Microdrive-Befehlen ist immer Drive 1 angesprochen

Stefan Arhes

MC-Generator

```
 10 REM ************************
 20 REM ** GRAF EDDI
 30 REM **   MC-Loader
 40 REM ** (C) 1985 by STARCOMP
 50 REM ** Tel.: 06021/97302
 60 REM ************************
 70
 80
 100 CLEAR 32767
 110 LET adr=62750
 120 DEF FN f(a$)=16*(CODE a$(b)
-48-39*(a$(b)>"9"))+CODE a$(b+1)
-48-39*(a$(b+1)>"9")
 130 FOR a=1000 TO 1290 STEP 10
 140 PRINT AT 10,0;"Bearbeite Ze
ile ";a
 150 LET sum=0
 160 READ a$,x
 170 FOR b=1 TO 100 STEP 2
 180 LET w=FN f(a$)
 190 LET sum=sum+w
 200 POKE adr,w
 210 LET adr=adr+1
 220 NEXT b
 230 IF x<>sum THEN  PRINT "Fehl
er in Zeile ";a: STOP
 240 NEXT a
 245 GO SUB 9050
 250 CLS : PRINT AT 4,0;"Alle Da
ten eingelesen."
 260 PRINT ''"Bitte loeschen Sie
dieses Pro    gramm nun mit NEW,
und laden Sie den BASIC-Teil von
GRAFIK-EDDI  ein."
 270 STOP
1000 DATA "310000ed4bfcffaf328bf
53e01328af5cd12f77476cdbcf5cd12f
776763ef6bc301c3ef6bd30203e09bc3
01b3e09bd30272e",5923
1005
1010 DATA "10bcc03e64bc20d601fff
fc97084febf30cc4710c97084fec030c
310f57905fef730094f10b0798530b41
0f73209f579fe09",6843
1015
1020 DATA "3a09f530ec10a5080000c
5e53ebf903020cdb02247043afe0f10f
d473a0af5f60020147aa0f53a0bf5f60
02007f1a02477e1",5782
1025
1030 DATA "c1c9f118f97e10f4219b9
bc53e7fdbfee6022006060800af710030
1ff01cde0f5c13ef7dbfee6012002266
4c93eefdbfef5e6",6035
1035
1040 DATA "04200168f1f5e60820016
0f1f5e610200161f1e601200326100c93
ef7dbfaa610200169c921009b1002210
0b6010010110040",4521
1045
```

```
1050 DATA "1ab61213230b78b120f6c
3baf601000321000ec1100ef1a3240f67
e123a40f67723130b78b120efc700210
0d111000001001b",4288
1055
1060 DATA "adb0c92100001100d1010
01badb0c921009b1100d101001b10092
100b61100d101001bcd2ef6cd41f6c30
6f7f306c0c3af0e",4873
1065
1070 DATA "ed4bfeffed5bfcff2afaf
f151daf320af53e9132c6f63afeff4f3
afaff6fc5444dcdc7f6c12803af18023
e01320bf5cd8cf5",7398
1075
1080 DATA "0d2d7bbd20a604257cba2
0d02100401100d101001badb0c900c5e
53abfcdac22f53ac6f69467f147047a0
710fde601b7e1c1",5018
1090 DATA "c93e01320af5af320bf5e
d4bfcffed5bfaff141c61cd8cf50c79b
b20f04c047ab820f2c900002100d1110
04001001bedb0c9",6119
1095
1100 DATA "c5cd8cf50c0c1605cd8cf
50c1520f9c1c50041605cd8cf5041520f
9c1c516050d0dcd8cf50d1520f9c1c50
0051605cd8cf505",5429
1105
1110 DATA "1520f9c1c93e01320bf5a
f320af5c3eaf6000000cd49f7ed4bfcf
fed5bfaff141c3e9132c6f6af320af52
afeff3afcff4f3a",7263
1115
1120 DATA "faff6fcdc7f6c52803af1
8023e01320bf5444dcd8cf5c10c2c79b
b20e604247ab820d8c3baf6ed4bfeffe
d5bfaff2afcff14",7206
1125
1130 DATA "1caf320af53e9132c6f63
afeff4f3afcff6fc5444dcdc7f6c1280
3af18023e01320bf5cd8cf50d2c7bbd2
0e604247cba20d8",6404
1135
1140 DATA "c3baf621004011009b010
01bedb0c9cd6b0ded4bfcffed5bfaff1
41caf320af53e9132c6f63afcff4fcdc
7f62803af18023e",6472
1145
1150 DATA "01320bf5cd8cf50c79bb2
0eb0470ba20e23ec93222f6cd0af6cdd
df73ec33222f6c92100b3cd3bf82100c
ecd3bf8c9010003",6375
1155
1160 DATA "3638230b78b120f8c7cd8
2f9cd06f7010000c57f0f0f0fe6e0a05
f79e618aa4057d82150ff06001a77231
410fad1c1c5d579",5823
1165
1170 DATA "0f0f0f4fe6e0a06f79e60
3ee58677af5e607cddaf06001158ffd
d2160ff1aa6dd77001322dd2310f5f1e
6380f0f0fcddaf0",6244
1175
1180 DATA "06001158ffdd2160ff1ae
affa6ddb600dd77001323dd2310f0d12
160ff06087e12142310fac1043a20b82
0820c06003e10b59",4971
1185
1190 DATA "c250f8c3baf6fe0120032
11af9fe022003212af9fe0320032122f
9fe0420032112f9fe0520032132f9fe0
62003213af9fe07",5085
1195
1200 DATA "20032142f9fe00c0214af
9c9aaaa5555aaaa5555aaff55ffaaff5
5ffaa7755eeaa7755aaaa55aa55aa55a
a55aa005500aa00",6605
1205
1210 DATA "5500aaaaaaaa555555550
000000000000000fffffffffffffff2
100e91100f2810003edb03e70003e682
100e906100e2077",4704
1215
1220 DATA "0057230d79f6007a20f50
500570e2070f6007a20eac306f72100f
21100e9010003edb0c306f7cd02f9cd0
6f7ed4b76facd25",5611
1225
1230 DATA "f578a1feff200e3ebfed4
376facdac22cddb0b10e52100581100f
2810003edb0c900000000ed4b76fa3e9
132c6f6af320af5",6100
1235
1240 DATA "cd25f5ed4376facd6b0d3
e02cd0116ed4b76fa78a1feffc070d61
63004faff202d57793274fac62030245
fcdc7f620043e20",6261
1245
1250 DATA "10023e8fc5d5d7d1c10c7
9bb20eb3a74fa4f0570ba20a2ed4374f
ac901ffffc906bf0e0079320af532c6f
621d1facdc7f620",6933
1255
1260 DATA "043e0118023e00772379c
6084f20ed70d60047d61030e5cd06f7e
d4b76fa70d6165779c6205f21d1fa7e3
20bf5cd8cf50c23",5620
1265
1270 DATA "79bb20f33a74fa4f0570b
a20eac3baf600000000ed4bfcff3afbf
f3c90676f32c1faed5bfaff5f3e9132c
6f6af320af5cdc7",7219
1275
1280 DATA "f62003af18023e01320bf
5c5d5c1cd8cf5c1041c2520e73ac1fa6
73afdff473afcff5f0c152d20d6c3baf
600af320bf53c32",6276
```

```
1285
1290 DATA "8af5ed4bfcffc312f7000
00000000000000000000000000000000
00000000000000000000000000000000
0000000000000000",1662
8000
8010
9000 SAVE "EDDI MC"
9010 VERIFY "EDDI MC"
9020 STOP
9030
9050 FOR a=USR "a" TO USR "a"+7
9060 READ s: POKE a,s: NEXT a
9070 DATA 66,60,66,66,126,66,66,
0,68,0,56,4,60,68,60,0,66,60,66,
66,66,66,60,0
9080 DATA 68,0,56,68,68,68,56,0,
66,0,66,66,66,66,60,0,68,0,68,68
,68,68,56,0
9090 DATA 56,68,68,72,68,68,72,6
4,255,255,255,255,255,255,255,25
5
9100 DATA 0,0,0,0,31,32,64,128,1
28,64,32,31,0,0,0,0,0,0,0,0
9110 DATA 248,4,2,1,1,2,4,248,0,
0,0,0,0,0,0,0,255,0,0,0,0,0,0
9120 DATA 255,0,0,0,0,128,128,12
8,128,128,128,128,128
9130 DATA 1,1,1,1,1,1,1,1,128,64
,32,31,31,32,64,128,0,0,0
9140 DATA 255,255,0,0,0,1,2,4,24
8,248,4,2,1
9150 RETURN
```

Hauptprogramm

```
   1 REM ***********************
   2 REM ** GRAF EDDI
   3 REM **  Hauptprogramm
   4 REM ** (C) 1985 by STARCOMP
   5 REM ** Tel.: 06021/97302
   6 REM ***********************
   7
  10 GO SUB 9000
  14 LET au=5000: LET x=1: LET y
=1: LET r=1
  15 LET uu=62750: LET xxyy=6553
0: LET xy=65532
  16 LET o=8: LET p=11: LET oo=5
: LET pp=10
  17 DIM b$(2,1): LET b$(1)="L":
 LET b$(2)="G"
  18 RANDOMIZE USR (uu+784)
  19 CLS : POKE xy,87: POKE xy+1
,87: LET oo=15: LET pp=10: PRINT
 "IMMMMMMMMK IMMMMMMMMK IMMMMMMM
MK"
  20 PRINT "Q";a$(1);"P Q";a$(2)
;"P Q";a$(3);"P"
  30 FOR a=4 TO 33 STEP 3
  40 PRINT "QRRRRRRRRS QRRRRRRRR
S QRRRRRRRRS"
  50 PRINT "Q";a$(a);"P Q";a$(a+
1);"P Q";a$(a+2);"P"
  60 NEXT a
  70 PRINT #0;AT 0,0;"JNNNNNNNNL
 JNNNNNNNNL JNNNNNNNNL"
  75
  80 REM *** Zeilen 19-70 UDGs !
  90
  91
 100 PRINT AT y,x; OVER 1; INVER
SE 1;" "
 110 IF INKEY$="8" OR CODE INKEY
$=9 THEN  IF x<>23 THEN  GO SUB
200: LET x=x+11: LET r=r+1: GO T
O 100
 120 IF INKEY$="5" OR CODE INKEY
$=8 THEN  IF x<>1 THEN  GO SUB 2
00: LET x=x-11: LET r=r-1: GO TO
 100
 130 IF INKEY$="6" OR CODE INKEY
$=10 THEN  IF y<>21 THEN  GO SUB
 200: LET y=y+2: LET r=r+3: GO T
O 100
 140 IF INKEY$="7" OR CODE INKEY
$=11 THEN  IF y<>1 THEN  GO SUB
200: LET y=y-2: LET r=r-3: GO TO
 100
 160 IF INKEY$<>"0" AND CODE INK
EY$<>13 THEN  GO TO 110
 170 RANDOMIZE USR (uu+488)
 180 GO TO 1000+r*100
 200 FOR z=1 TO 20: NEXT z: PRIN
T AT y,x;a$(r): RETURN
1100 POKE 23607,235: GO TO au
1200 POKE 23607,238: GO TO au
1300 POKE 23607,60: GO TO au
1400 GO SUB 8200
1410 RANDOMIZE USR (uu+1370)
1420 GO TO au
1500 GO SUB 8000
1510 GO SUB 8100
1520 RANDOMIZE USR (uu+350)
1530 GO TO au
1600 GO SUB 8000
1610 GO SUB 8100
1620 RANDOMIZE USR (uu+638)
1630 GO TO au
1700 GO SUB 8000
1710 GO SUB 8100
1720 RANDOMIZE USR (uu+573)
1730 GO TO au
1800 GO SUB 8000
1810 GO SUB 8100
1820 RANDOMIZE USR (uu+567)
1830 GO TO au
1900 GO SUB 8000
1910 RANDOMIZE USR (uu+555)
```

```
1920 RANDOMIZE USR (uu+412)
1930 GO TO au
2000 GO SUB 8000
2010 RANDOMIZE USR (uu+715)
2020 GO TO au
2100 GO SUB 8000
2110 RANDOMIZE USR (uu+451)
2120 RANDOMIZE USR (uu+412)
2130 GO TO au
2200 LET xx=USR (uu+1183): IF xx
=65535 THEN  GO TO 19
2215 PRINT #0;AT 1,0;"STORE
                 EXIT"
2220 LET bu=0: PRINT AT oo,pp; O
VER 1; FLASH 1;"■": GO SUB 7000
2225 IF oo=22 THEN  GO TO 2250
2230 IF bu THEN  PRINT AT oo,pp;
 OVER 1;"■"
2240 GO TO 2220
2250 LET oo=21
2260 IF pp>=20 THEN  GO TO au
2270 IF pp<=3 THEN  RANDOMIZE US
R (uu+1202): GO TO au
2280 GO TO 2220
2300 CLS : PRINT AT 5,10;"IMMMMM
MMMK";AT 6,10;"QATTRIBUTP";AT 7,
10;"QRRRRRRRRS";AT 8,10;"QCHESS
  P";AT 9,10;"QRRRRRRRRS"
2310 PRINT AT 10,10;"QSET ATTRP"
;AT 11,10;"QRRRRRRRRS";AT 12,10;
"QEXIT    P";AT 13,10;"JNNNNNNNN
L"
2320 PRINT AT o,p; OVER 1; INVER
SE 1;"          ": FOR z=1 TO 20:
NEXT z
2330 IF INKEY$="6" OR CODE INKEY
$=10 THEN  PPINT AT o,p; INVERSE
 1; OVER 1;"          ": LET o=o+2
*(o<>12): GO TO 2320
2340 IF INKEY$="7" OR CODE INKEY
$=11 THEN  PRINT  INVERSE 1; OVE
R 1;AT o,p;"          ": LET o=o-2
*(o<>6): GO TO 2320
2345 IF INKEY$="0" OR CODE INKEY
$=13 THEN  GO TO 2360+o*2
2350 GO TO 2330
2372 RANDOMIZE USR (uu+1124): GO
 TO au
2376 RANDOMIZE USR (uu+1076): GO
 TO 5000
2380 GO SUB 6000
2382 RANDOMIZE USR (uu+1138): RA
NDOMIZE USR (uu+1171): GO TO 230
0
2384 GO TO 19
2400 RANDOMIZE USR (uu+303)
2410 RANDOMIZE USR (uu+408)
2420 RANDOMIZE USR (uu+1171)
2430 GO TO au
2500 RANDOMIZE USR (uu+291)
2510 GO TO au
2600 RANDOMIZE USR (uu+703)
2610 GO TO au
2700 INK 9: PAPER 8: LET st=0: L
ET gr=0
2710 LET f10=1: GO SUB 7200
2720 GO SUB 7100
2730 IF st=1 THEN  RANDOMIZE USR
 (uu+412): GO TO 19
2740 GO TO 2720
2800 RANDOMIZE USR (uu+263)
2810 GO TO au
2900 RANDOMIZE USR (uu+236)
2910 GO TO au
3000 RANDOMIZE USR (uu+241)
3010 GO TO au
3100 RANDOMIZE USR (uu+889)
3110 GO TO au
3200 RANDOMIZE USR (uu+315)
3210 RANDOMIZE USR (uu+1171)
3220 GO TO au
3300 RANDOMIZE USR (uu+326)
3310 RANDOMIZE USR (uu+1171)
3320 GO TO au
3400 RANDOMIZE USR (uu+344)
3410 GO TO au
3500 COPY : REM Hier kann der Ma
schinencodeaufruf Ihrer Hardcopy
routine stehen
3510 GO TO au
3600 GO SUB 5200
3610 CLS : SAVE n$CODE 53504,691
2
3620 GO TO au
3700 GO SUB 5200
3710 CLS : SAVE *"m";1;n$CODE 53
504,6912
3720 GO TO au
3800 CLS : CAT 1: GO TO au
3900 GO SUB 5200
3910 CLS : LOAD n$CODE 32768,691
2
3920 GO TO 2400
4000 GO SUB 5200
4010 LOAD *"m";1;n$CODE 32768,69
12
4020 GO TO 2400
4100 GO SUB 5200
4110 ERASE "m";1;n$
4120 GO TO au
4200 GO SUB 5100
4210 CLS : LOAD n$CODE 60416,768
4220 GO TO 4320
4300 GO SUB 5100
```

```
4310 CLS : LOAD *"m";1;n$CODE 60
416,768
4320 RANDOMIZE USR (uu+488)
5000 IF INKEY$<>"" THEN  GO TO a
u
5010 IF INKEY$="" THEN  GO TO 50
10
5020 GO TO 19
5100 CLS : PRINT "Name des Chara
ktersatzes:"
5110 INPUT AT 20,20;AT 0,10; LIN
E n$
5120 IF LEN n$=0 THEN  GO TO 19
5130 RETURN
5200 CLS : PRINT "Name des Bilde
s:"
5210 INPUT AT 20,20;AT 0,10; LIN
E n$
5220 IF LEN n$=0 THEN  GO TO 19
5230 RETURN
6000 INPUT "Paper: ";pa
6010 IF pa<0 OR pa>7 THEN  GO TO
 6000
6020 INPUT "Ink: ";in
6030 IF in<0 OR in>7 THEN  GO TO
 6020
6040 INPUT "Flash: ";fl
6050 IF fl<0 OR fl>1 THEN  GO TO
 6040
6060 INPUT "Bright: ";br
6070 IF br<0 OR br>1 THEN  GO TO
 6060
6080 POKE 23696,0: POKE 23695,12
8*fl+64*br+8*pa+in
6090 RETURN
7000 PAUSE 0
7005 PRINT AT oo,pp; FLASH 0; OV
ER 1;"█"
7010 IF INKEY$="5" OR CODE INKEY
$=8 THEN  LET pp=pp-1*(pp<>0)
7020 IF INKEY$="8" OR CODE INKEY
$=9 THEN  LET pp=pp+1*(pp<>31)
7030 IF INKEY$="6" OR CODE INKEY
$=10 THEN  LET oo=oo+1
7040 IF INKEY$="7" OR CODE INKEY
$=11 THEN  LET oo=oo-1*(oo<>0)
7050 IF INKEY$="0" OR CODE INKEY
$=13 THEN  LET bu=1
7060 RETURN
7100 LET z$=INKEY$: LET zz=CODE
z$
7105 IF z$="" OR zz<8 OR (zz>11
AND zz<32 AND zz<>15) THEN  GO T
O 7100
7106 BEEP .01,10: LET f10=0: GO
SUB 7200: LET f10=1
7110 IF zz=15 THEN  LET gr=ABS (
gr-1): GO TO 7200
```

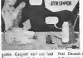
```
7120 IF zz=8 THEN  LET pp=pp-1*(
pp>0): GO TO 7200
7130 IF zz=9 THEN  LET pp=pp+1*(
pp<31): GO TO 7200
7140 IF zz=11 THEN  LET oo=oo-1*
(oo>0): GO TO 7200
7150 IF zz=10 THEN  LET oo=oo+1*
(oo<23): GO TO 7200
7160 IF z$=" STOP " THEN  LET st
=1: RETURN
7180 IF zz>127 THEN  GO SUB 7200
: GO TO 7100
7190 GO TO 7300
7200 IF oo>21 THEN  GO TO 7230
7210 PRINT AT oo,pp; OVER 1; FLA
SH f10;b$(gr+1)
7220 RETURN
7230 PRINT #0; PAPER 8; INK 8;AT
 oo-22,pp; OVER 1; FLASH f10;b$(
gr+1)
7240 RETURN
7300 IF gr=1 THEN  GO TO 7400
7310 IF oo>21 THEN  GO TO 7350
7320 PRINT AT oo,pp;CHR$ zz
7330 LET pp=pp+1*(pp<31): GO TO
7200
7350 PRINT #0;AT oo-22,pp; INK 8
; PAPER 8;CHR$ zz: GO TO 7330
7400 IF zz<97 OR zz>117 THEN  GO
 SUB 7200: GO TO 7100
7410 LET zz=zz+47: GO TO 7310
8000 LET xx=USR uu: BEEP .1,10:
IF xx=65535 THEN  GO TO 19
8010 LET x1=INT (xx/256): LET y1
=xx-x1*256: POKE xy,y1: POKE xy+
1,x1: RANDOMIZE USR (uu+1444)
8020 LET xx=USR uu: IF xx=65535
THEN  GO TO 19
8030 LET x2=INT (xx/256): LET y2
=xx-x2*256
8040 IF x2<=x1 OR y2<=y1 THEN  G
O TO 8020
8050 RANDOMIZE USR (uu+1444): PO
KE xxyy,y2: POKE xxyy+1,x2
8060 RETURN
8100 BEEP .1,10: LET xx=USR uu:
BEEP .1,10: IF xx=65535 THEN  GO
 TO 19
8110 LET x1=INT (xx/256): LET y1
=xx-x1*256
8120 POKE 65534,y1: POKE 65535,x
1: RETURN
8200 LET xx=USR uu: BEEP .1,10:
IF xx=65535 THEN  GO TO 19
8210 LET x1=INT (xx/256): LET y1
=xx-x1*256: POKE xy,y1: POKE xy+
1,x1: RANDOMIZE USR (uu+1444)
8220 LET xx=USR uu: IF xx=65535
THEN  GO TO 19
```

```
8230 LET x2=INT (xx/256): LET y2
=xx-x2*256
8240 IF x1>=x2 THEN  GO TO 8220
8250 RANDOMIZE USR (uu+1444): PO
KE xxyy,y2: POKE xxyy+1,x2
8260 RETURN
9000 RESTORE : DIM a$(33,8)
9010 FOR a=1 TO 33: READ a$(a):
NEXT a
9020 DATA "CHAR 1","CHAR 2","NOR
MCHAR","ROTATE","MIRROR H","MIRR
OR V","COPY","MOVE","DELETE","FE
TCH","INVERSE","ZOOM"
9030 DATA "ATTR","RECALL","BACKU
P","LD 1--8","PRINT","EX CHAR","
OR S--1","OR S--2","SCHATT","EX
S--1","EX S--2","SINCLAIR"
9040 DATA "EIGEN","SAVE","MSAVE"
,"CATALOG","LOAD","MLOAD","ERASE
","LOAD CH","MLOAD CH"
9050 RETURN
9800
9810
9820 SAVE "EDDI" LINE 9999
9830 SAVE "EDDICODE"CODE 62750,1
500: SAVE "EDDI UDG"CODE USR "a"
,168
9840
9850 VERIFY "EDDI": VERIFY "EDDI
CODE"CODE : VERIFY "EDDI UDG"COD
E
9860 STOP
9870
9880 SAVE "EDDI": VERIFY "EDDI"
9890 STOP
9900
9999 CLEAR 32767: LOAD "EDDICODE
"CODE 62750: LOAD "EDDI UDG"CODE
 USR "a": RUN
```

Poke Stripper

Nach "Strip Poker", "Peepshow" und "Samantha Fox Strip-Poker" liegt mit "Poke Stripper" nun ein weiteres Programm vor, in dem unbekleidete Damen eine Rolle spielen.

Das Programm bietet dem Spieler eine handfeste Pokerpartie. Gespielt wird mit fünf Karten, die grafisch gut dargestellt sind. Folgende Wertungen werden gezählt:


Handfester Poker um Schlips und Kragen

Straight Flush – Straße aus einer Farbe

Four of a kind – 4 Karten eines Wertes

Full House – ein Paar und ein Drilling

Flush – alle Karten haben die gleiche Farbe

Straight – eine Sequenz von Werten

Triplet – ein Drilling

Two Pairs – zwei Paare

One Pair – ein Paar

Beide Spieler (also ein Spieler und der Spectrum) erhalten zum Programmstart ein Kapital von 100 $. Vor jeder Runde kommen davon 10 $ in den Pott. Danach können maximal 3 Karten neu ausgegeben werden, um das bestmögliche Blatt zu erreichen. Nun kann man den Einsatz stehenlassen oder erhöhen. Bis zu diesem Punkt handelt es sich also um eine ganz normale Pokerpartie.

Hat nun aber das auf dem Bildschirm dargestellte Mädchen nicht mehr genug Kapital, um mitzugehen, muß es ein Kleidungsstück ablegen. Das soll dem Programm die nötige Würze geben. Insgesamt bietet "Poke Stripper" drei Mädchen. Hat man sie entblättert, endet das Spiel.

Da die Grafik ansprechend gestaltet und auch eine Spielmotivation vorhanden ist, kann man das Spiel eigentlich als gut bezeichnen. Ich bin einmal gespannt, ob in Zukunft vielleicht ein Programm auf den Markt kommt, in dem Männer ausgezogen werden – natürlich nur im Rahmen der Gleichberechtigung.

System Spectrum 48 K
Hersteller Individual Software

Rolf Knorre

30 Games

Aus England erreicht uns ein Leckerbissen für Spectrum-Einsteiger. Es handelt sich um ein Software-Paket mit insgesamt 30 Spielen. Allerdings sind es keine neuen Titel, sondern leicht angestaubte Programme, die alle schon mal in den letzten drei Jahren veröffentlicht wurden. Ihre Qualität reicht von hervorragend bis eher mittelmäßig. Trotzdem kann man die Sammlung denjenigen, die bisher noch keine Programme haben, durchaus empfehlen. Da der Preis nicht zu hoch ist, wird sicher jeder auf seine Kosten kommen. Folgende Titel sind enthalten:

"Carpet Capers", "Antics", "Planetfall", "Gridrunner", "Bismark", "Meteor Storm", "Jet Set Gertie", "3D Tunnel", "Stockmarket", "Fudge Frenzy", "The Valley", "Dragonsbane", "Demon Knight", "Nede Garden", "Space Intruders", "Mighty Magus", "Fall of Rome", "Draughts", "Xadom", "Detective", "20 Tons", "Chessplayer", "Micromouse", "Tubecube", "Invasion", "Escape", "Lazer Zone", "Strontium Dog", "Pyromania", "Blood 'n Guts".

Wer seinen Spectrum schon länger benutzt, wird sicher das eine oder andere Programm noch kennen. Mit wenigen Ausnahmen handelt es sich um Action-Spiele aller Art. Ausgeliefert wird "30 Games" auf zwei Cassetten mit den entsprechenden Spielbeschreibungen.

System. Spectrum 48 K
Verschiedene Hersteller

Rolf Knorre

3 kurze Programme für den ZX Spectrum

„Elite-Code" ist die Auflösung des Spiels „Elite". Der von diesem Programm erstellte Code wird mit dem LOAD-Menüpunkt bei „Elite" geladen. Nach dem Starten des Spiels braucht man nur noch ein Objekt abzuschießen, um zur sagenumwobenen Elite des Weltraums zu gehören. Das Programm sichert den Code nach dem Start normal ab, wobei man darauf achten muß, daß der Header, der beim Laden sonst den Namen auf dem Bildschirm hervorruft, nicht auf Cassette abgespeichert wird, d.h. man darf den Recorder erst nach dem Header anschalten.

Das Programm „Streckung" verbreitert Text und Grafik auf dem Bildschirm. Nach jedem Aufruf mit RANDOMIZE USR 30000 wird alles in Sekunden um ein Pixel breiter. Der Code wird automatisch abgespeichert und überprüft (VERIFY), worauf ein kleines Demo folgt. Die Routine ist übrigens auf jede beliebige Adresse verschiebbar.

Das Programm „Primzahlenberechnung" erstellt auf einem Spectrum 48 K theoretisch bis zu 8000 Primzahlen. Die Zahlen werden relativ schnell ermittelt und im Array „A" gespeichert. Die Dauer der Berechnung nimmt mit dem Ansteigen der Primzahlen zu, da es nun immer mehr gibt, durch die eine Zahl geteilt werden muß, damit der Computer weitere Primzahlen feststellen kann.

Gunter Imayer

Elite-Code

```
   1 REM     ELITE-CODE
   2 REM  von Gunther Imayer
   3
  10 FOR n=3e4 TO 3e4+22: READ a
: POKE n,a: NEXT n
  20 DATA 69,76,73,84,69,0,82,50
,48,48,0,64,255,255,25,14,0,7,19
9,255,97,11,150
  30 FOR n=3e4+23 TO 3e4+39: POK
E n,0: NEXT n
  40 FOR n=3e4+40 TO 3e4+102: RE
AD a: POKE n,a: NEXT n
  50 DATA 35,3,1,0,0,0,70,4,69,1
27,0,0,255,255,255,255,255,255,0
,0,82,210,66,16,154,109,106,69,5
0,172,74,161,5,6,16,17,23,25,34,
15,0,0,15,41,0,39,60,13,31,19,0,
0,1,206,154,2,92,212,137,243,240
,149,0
  60 SAVE "ELITE"CODE 3e4,102
```

Streckung

```
  10 LET a=0: FOR n=3e4 TO 30028
: READ b: LET a=a+b: POKE n,b: N
EXT n: IF a<>2671 THEN  PRINT "F
ehler in Data !!": STOP
  20 DATA 55,69,17,0,24,33,0,64,
120,31,35,27,56,11,43,102,119,35
,122,179,254,0,32,240,201,203,25
4,24,241
  30 SAVE "Streckung"CODE 3e4,30
: VERIFY ""CODE
  40 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS : PRINT "D E M O"
  50 FOR n=-PI TO PI STEP .2: PL
OT 43,80: DRAW 170,0,n: NEXT n:
FOR n=-PI TO PI STEP .2: PLOT 12
0,6: DRAW 0,160,n: NEXT n: PAUSE
 5: PAUSE 100
 100 FOR n=0 TO 30: RANDOMIZE US
R 30000: NEXT n
```

Primzahlenberechnung

```
   1 REM Primzahlenberechnung
   2 REM  von Gunther Imayer
   3
  10 DIM a(8000): LET a(1)=2
  20 LET b=1
  30 FOR n=2 TO 1e20
  40 FOR c=1 TO b/2+1
  50 IF n/a(c)<>INT (n/a(c)) THE
N  NEXT c: LET b=b+1: LET a(b)=n
: PRINT AT 0,0;n: NEXT n
  70 NEXT n
```

Tantalus

Wieder einmal ist die Menschheit in Gefahr, wieder einmal droht ihr die totale Vernichtung – die Punkoiden greifen an.

Das neueste Programm von Quicksilva bietet eine im Science-fiction-Milieu angesiedelte Schlacht apokalyptischen Ausmaßes. Wer sich für Action-Schießspiele begeistern kann, wird hier gut bedient. In einem technologisch gestylten Labyrinth muß der Spieler einen kleinen Roboter auf die Jagd nach den Punkoiden schicken, die in vielen Formen und ungeheurer Zahl auftreten. Da man nicht gleich bei jeder Kollision ein Leben verliert, macht die Ballerei eine Menge Spaß, zumal die Grafik wirklich hervorragend ist. Nur ist leider kein Sound vorhanden. Trotz der einfachen Geschichte wird „Tantalus" viele Freunde finden.

System	Spectrum 48/128 K
Hersteller	Quicksilva

Rolf Knorre

SPECTRUM ✳ Hard- und Software zu günstigen Preisen ✳ SPECTRUM

Das neue ISO-ROM für das Spectrum! Jetzt noch besser! Mit:
- **Reset ohne Programmverlust!** • **Deutsche Umlaute!**
- **Reset Zeichensatz** • **Verbesserten CAT für MD!**
- **Verkürzte Syntax für Microdrive oder Beta-Disk!**
- **Erweiterter Editor mit Cursor(Hoch Programmzeilen!)**
- **Variable Zeichenbreite 32, 36, 42, 64 oder 64 Zeichen pro Zeile** nur bei Monitor-ROM)

Dazu eine von 2 **NMI-Routinen**

Entweder der HEX-Monitor: Unterbricht jedes Programm, ermöglicht Schritt hier und Starten eigener MC-Routinen
Oder die neue Backup-Routine: Überträgt sämtliche Programme auf Ihr Speichermedium. Dabei werden nur die **belegten Bytes abgespeichert**, also nicht immer volle 48 K! Die übertragenen Programme laufen natürlich auch ohne ISO-ROM

Außerdem: **POKE-Option!** Sie können jederzeit Ihr Game unterbrechen, einen POKE eingeben und weiterspielen.
Unentbehrlich für alle Spieler!!
Backup-Routinen gibt's für Microdrive, Opus Disk, Beta-Disk und Cassette (für alle anderen Speicher [illegible]) **Fordern Sie zum ISO-ROM unser Info an!**

Komplett mit deutscher Anleitung und Einbauplan für Einbau direkt im Computer (auch für Parallelbetrieb mit Sinclair-ROM)
Monitor-ROM: Jetzt nur 79,– DM **Backup-ROM: Jetzt nur 90,– DM**
Auf Wunsch bauen wir Ihr ROM für nur 20,– DM in Ihren Computer ein.

SYS: Das vollständige MC-Entwicklungspaket für den Spectrum! Löst alle MC-Probleme. Superschneller **Macroassembler** mit eigenem platzsparendem Textformat. **Disassembler, Debugger** mit Single-Step und Trace und **symbolischem Reassembler!** Unterstützt alle Speichermedien (bitte bei Bestellung angeben!) **Einführungspreis nur 90,– DM**

Amadeus: Das Spectrum-Musiksystem für alle 3-Kanal-Soundmodule. **Synthesizerteil** mit 7 Instrumenten. **Notenleser** für bequeme Noteneingabe und Korrektur. **Jetzt nur 45,– DM**
Zusammen mit 3-Kanal-Soundmodul nur 125,– DM

Jetzt brandneu: Das ISO-FACE! Ermöglicht den Betrieb von bis zu 3 weiteren ROMs durch einfaches Aufstecken an den Spectrum
- **3 Steckplätze für 27128-er EPROMs!**
- **Umschalter Steckplatz/Int. ROM absturzfrei.** 3-fach Schalter für den Int. ROM und getrennte Reset/NMI-Taste
- **Durchgeführter Bus und stabiles Gehäuse!**

Das alles für sage und schreibe nur 100,– DM!
ISO-FACE komplett mit einem ISO-ROM nur 130,– DM!

BETA PACK – Das Utility-Paket für Ihre Beta-Disk
1. **BETA TRANS** – Bringt Files von Cassette auf Diskette (auch headerless)
2. **BETA COPY** – Kopiert beliebige Files von Disk zu Disk!
3. **BACKUP 48** – Schnelles Backup, überträgt alle 48 K auf einmal
4. **BETADRESS** – Findet Anfangsadressen von headerlosen Codeblöcken
5. **BETA TAPE** – Überträgt Diskettenprogramme zurück auf Cassette!

Alle Programme arbeiten mit einem oder mehreren Laufwerken und allen DOS-Versionen! (Bitte angeben)
5 Topprogramme auf 1 Diskette: Nur 70,– DM

ISO-DATEI – Endlich ein Dateiprogramm, das den Speicherplatz der Beta-Disk voll ausnutzt. Bis zu 5000 Einträge auf einer Disk. [illegible] deutsche Umlaute. Programm und Anleitung in Deutsch, sehr benutzerfreundlich.
Jetzt nur 45,– DM

BETA BOOT – „Das" File für jede Diskette! Zum Laden, Umbenennen, Löschen usw. nur noch File per Cursor anfahren. **Nur 25,– DM**

POKE-STRIPPEN – Setzen Sie Ihr Pokerface auf und ziehen Sie sich warm an! Schaffen Sie es, gegen Sheila, [illegible] und Diane Sieger zu bleiben? Das Spectrum-Strippoker mit den [illegible] Sprüchen und den heißesten Mädchen!
Nur 30,– DM

KEMPSTON PRO – Das Universal-Joystick-Interface für den Spectrum! Kempston + Sinclair + **Cursor-Interface in einem. Jetzt nur 60,– DM**

Gratis-Info und Bestellungen (V-Scheck oder Nachnahme) bei:

Volker Marohn
Am Beilstück 30
4600 Dortmund 50

INDIVIDUAL SOFTWARE

Telefonische Bestellungen:
0231/71 66 68

ZX Spectrum Hardware

Verlag Markt & Technik
Von R. Egeler
150 Seiten, DM 29,80
ISBN 3-89090-092-5

Dieses Buch gibt allen Hardwarefreaks und solchen, die es werden wollen, eine optimale Hilfestellung für ihre Basteleien in und am Spectrum. Daneben ermöglichen die vermittelten Kenntnisse ein leichteres Auffinden von Hardwarefehlern.

Das Buch gliedert sich in zwei Teile. Zunächst werden die verschiedenen internen Baugruppen des Spectrum erläutert. Begonnen bei der Z80-CPU, der Speicherorganisation, den vielfältigen Aufgaben des ULA usw. werden alle Baugruppen ausführlich anhand von Signalgebung und Schaltungsauszügen erklärt. Die Beschreibung des Spectrum-Inneren ist so gewählt, daß jeder Bereich erfaßt wurde. Daneben werden in einem gesonderten Kapitel Tips für die Fehlersuche bei einem Defekt gegeben. Hinweise auf die Anschaltung von Interface 1, welche man im Notfall aber auch einem der im Handel befindlichen Microdrive-Bücher entnehmen kann, fehlen leider. Im Anfang des Buches befinden sich noch zwei herausklappbare Schaltpläne der Spectrum-Versionen Issue 2 und 3, einschließlich der Unterschiede zu Issue 3b.

Der zweite und umfangreichere Teil des Buches befaßt sich mit eigenen Hardwareerweiterungen, die ohne großen Aufwand nachgebaut werden können. Alle in den Basteleien vorkommenden Sonderbausteine werden – wie schon bei der Besprechung der Sinclair Hardware – anhand ihrer Anschlüsse und Funktionen genau erläutert. Der Reigen der Basteleien beginnt mit einer Einführung in die möglichen Adressierungsarten der I/O-Geräte und der Anleitung zum Aufbau einer PIO (parallel Input/Output), mit der dann die nachfolgenden Basteleien ohne Probleme oder Gefahren für den Spectrum angesteuert werden. Neben einigen kleinen Bausteinen für Ein- und Ausgabe mit der PIO und einer Erklärung der PIO-Programmierung in Maschinensprache und Interrupts wurden vom Autor R. Egeler noch drei große Bauanleitungen aufgezeichnet und dokumentiert.

Da ist zunächst die Anleitung zum Bau einer softwaregesteuerten Hardwareuhr, die dank eines Akkus auch noch nach dem Ausschalten der Spectrum-Stromversorgung weiter läuft. So kann man z. B. die aktuelle Uhrzeit abfragen, um diese in einem Programm weiterzuverwenden.

Wer sich an die zweite Bauanleitung, einen Soundgenerator mit dem bewährten IC AY-3-8912 wagt, kann eine Platine aufbauen, mit welcher der Spectrum in Sachen Musik nun auch seinen Konkurrenten Commodore und Schneider nicht mehr nachhinkt, da diese mit demselben IC bestückt sind. Neben einer kompletten Beschreibung des Platinenaufbaus und der Funktionsweise des Sound-ICs ist auch das Listing eines komfortablen Musik-Editors abgedruckt.

Da ja bekanntlich die Herstellung der Platine, das Beschaffen aller Bauteile oder Abtippen der Listings manchmal sehr umständlich ist, kann man im Anhang des Buches eine Bezugsquelle erfahren, über die Bausätze, Platinen oder Programme zu beziehen sind.

Die letzte Anleitung befaßt sich mit dem Aufbau eines Analog/Digital-Wandlers. Dieser wird im Buch als Speicheroszilloskop für den Spectrum verwendet, eignet sich aber auch für andere Zwecke der Meßwerterfassung.

Nach dieser kleinen Inhaltsbeschreibung kann wohl jeder Interessierte selbst entscheiden, ob sich der Kauf des Buches lohnt. Zusammenfassend kann man wohl sagen, daß jeder, der mehr aus seinem Spectrum machen will, mit diesem Buch sehr gut beraten ist. Auch ist der Preis von unter 30,– DM für eine derartige Ansammlung von Know-how ein weiterer Pluspunkt.

Thomas Bettelde

Quercopy für Spectrum 16/48 K

Jetzt können Sie nicht nur sehr lange Ausdrucke anfertigen, sondern auch sehr breite. "Quercopy" erstellt eine Bildschirmkopie im Querformat (24 Zeilen!)

Selbstverständlich ist die kurze MC Routine nicht an einen bestimmten Speicherplatz gebunden. Aufgerufen wird sie mit der Ladeadresse ("RANDOMIZE USR xxxxx"), wonach die 32 Spalten hintereinander quer ausgedruckt werden.

Insbesondere Benutzer des Seikosha GP 50 S werden sich über den Querausdruck freuen, da der Nadelkopf nicht quadratisch ist und man jetzt zwischen zwei "verzerrten" Bildern das bessere auswählen kann. Metallpapier-Freunde mit einer 48 K Version werden es zu schätzen wissen, daß mittels Einstieg über "RANDOMIZE USR xxxxx+5" erst das gesamte Querbild erzeugt und danach in einem Durchgang ohne Stoppen des Motors abgedruckt wird. Das gelingt aber nur, wenn das Programm nicht im Bereich 39000-47192 steht. Falls dieser auf yyyyy bis yyyyy+8179 verschoben werden soll, sind das 7 (LOW-) bzw. 8 (High-) Byte (hier 88 und 151) entsprechend mit der Adresse yyyyy−256 zu ändern.

Nach Eintippen und RUN speichern Sie das Programm bitte mit SAVE "quercopy" CODE 30000 152 ab.

Harald Hagemann

Quercopy

```
  10 REM <quercopy> , h.h.
  20 CLEAR 29999: RESTORE 100
  30 LET fst=30000: LET sum=0: F
OR i=0 TO 151: READ j: LET sum=s
um+j: POKE fst+i,j: NEXT i: IF s
um<>15520 THEN  PRINT "error!"
 100 DATA 1,0,0,24,3,1,88,151,19
7,33,59,92,203,206,205,223,14,21
7,22,32,217,33,0,64,217,14,0,217
,221,33,27,91,217,6,3,217,14,0
 110 DATA 6,0,203,6,221,203,0,30
,221,43,17,32,0,25,16,242,17,0,0
,221,25,13,32,232,17,246,255,221
,25,217,5,217,32,220,17,0,232,25
 120 DATA 17,56,0,221,25,217,13,
217,32,202,241,245,254,0,32,41,2
29,205,205,14,225,35,217,21,217,
32,177,241,234,0,40,19,243,22,0,
33,88,152
 130 DATA 6,10,205,244,14,33,120
,152,6,235,205,211,14,33,59,92,2
03,142,201,235,193,33,0,1,9,229,
213,235,33,0,91,1,0,1,237,176,24
,200
```

Low Price Software für den Spectrum

Ein Trend in der Softwarebranche ist nicht zu übersehen: Die Programme werden billiger. Die Firma Mastertronic war einst Auslöser dieser Welle. Mit Programm-Cassetten um 10 DM hat sie sich inzwischen auf dem Markt etablieren können. Nachahmer gibt es auch schon. Dabei hat sich herausgestellt, daß preiswerte Software nicht unbedingt schlecht sein muß.

Das englische Softwarehaus Elite Systems Ltd., bekannt durch Titel wie "Frank Bruno's Boxing", "Commando" oder "Ghosts n Goblins", hat vor einigen Wochen ein Label aus der Taufe gehoben, das dem neuen Trend folgen soll. Unter der Bezeichnung 2.99 Classics wurden aber keine neuen Titel aufgelegt, sondern ausschließlich ältere, bereits mit Erfolg verkaufte Programme angeboten. Besonders interessant ist der Verkaufspreis, der sich schon aus der Label-Bezeichnung ableiten läßt: 2.99 steht für 3.99 englische Pfund, was etwa 11,- DM entspricht. Folgende Titel sind bisher für Spectrum Besitzer erhältlich:

"Full Throttle", "3D Death Chase", Skool Daze, "Valhalla", "Fool", "Scuba Dive", "Chess", "3D Star Strike", "Jasper", "3D Tank Duel", "Tornado Low Level".

Damit können besonders Neueinsteiger preiswert zu Programmen kommen, die zwar schon älter sind, aber dennoch hervorragende Unterhaltung bieten. Die Reihe soll in nächster Zeit ständig erweitert werden. Auch andere Computertypen werden dann Berücksichtigung finden. Ein deutscher Lieferant war zum Zeitpunkt der Drucklegung noch nicht bekannt, wird aber sicher nicht mehr lange auf sich warten lassen.

Rolf Knorre

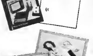
Spectrum
User Club
Wuppertal

Informationen erhalten Sie (gegen
Einsendung von DM 0.50 Rückporto) von:
Rolf Knorre,
Postfach 20 01 02, 5600 Wuppertal 2

it's sinclair time

Die drei Superspiele zum Wahnsinnspreis von je DM **24,-**

2 Bücher unentbehrlich - ehrlich !
ZX 81 DM **49,-**
Sinclair QL DM **69,-**



Das schärfste für den Spectrum - Peep Show
Bitte Alter angeben
DM **29,90**

Actiongame mit
32 Screens
32 Screens 100% Maschinensprache
DM **19,-**

Eine Reise von New York nach Ägypten. Deutsches Grafik-Adventure für DM **16,-**

Spezialitäten-Bestellschein

Anzahl	Best.-Nr.	Artikel	
	N8	Paperboy	
	N9	Light Force	
	N3	Alles über den ZX81	
	N4	QL unter Kontrolle	
	N5	Peep-Show	
	N6	Superstory	
	N7	G Force	
	N10	1942	

Ich wünsche folgende Bezahlung
☐ Nachnahme
[illegible]
☐ Vorauskasse
[illegible]

Name des Bestellers ____________ Name ____________

Anschrift ____________ Datum, Unterschrift ____________

[illegible]

Computer Kontakt, Softwareversand, Postfach 16 40, 7518 Bretten.

Zitterschrift

Bei "Zitterschrift" für den Spectrum 16/48 K handelt es sich um ein kurzes Programm, mit dem man ein Wort (Satz) auf dem Bildschirm "zittern" lassen kann. Die Variablen haben folgende Bedeutung:

po1 Position des Wortes auf dem Bildschirm
po2 Position des Wortes auf dem Bildschirm
le Länge des Wortes
l$ das Wort selbst

Es ist zu beachten, daß man, um aus dem Programm herauszukommen, nicht Break drücken darf, sondern irgendeine andere Taste, weil das Programm den Zeichensatz in der Adresse 23606 verändert!

Torsten Rinne

```
  5 LET po1=10: LET po2=11: REM
    POSITION DES WORTES
 10 LET x=0
 20 LET p=0
 30 LET a$=""
 40 LET le=7: REM
    LAENGE DES WORTES
 50 DIM l$(le)
 60 REM UMWANDLER
 70 LET l$="T.RINNE": REM
    DAS WORT WELCHES ZITTERT
 80 FOR n=1 TO le
 90 LET a$=a$+CHR$ ((CODE l$(n)
)-1)
 100 NEXT n
 105 REM ZITTERN
 110 PRINT AT po1,po2;a$
 120 POKE 23606,x+7
 130 IF p=0 THEN  LET x=x+1: IF
x=3 THEN  LET x=x-1: LET p=1
 140 IF p=1 THEN  LET x=x-1: IF
x=-1 THEN  LET x=x+2: LET p=0
 150 IF INKEY$<>"" THEM  POKE 23
606,0: STOP
 160 GO TO 110
```

Core

"Core" ist ein neues Programm aus dem bei Programmierern so beliebten Science-fiction-Milieu. Die Story könnte jedenfalls Grundlage für einen interessanten Film sein. Commander Andrew Angello erhält den Auftrag, eine von der Erde installierte wissenschaftliche Station auf dem Planeten EROC I zu untersuchen, da man von ihr Notsignale aufgefangen hat. Als Angello per Teleporterstrahl dort eintrifft, lebt von den 720 Wissenschaftlern niemand mehr. Seine Aufgabe ist es nun, versteckte Teile eines Computers zusammenzutragen, um mit diesem einen Gegenangriff gegen die Invasoren zu starten.

Der Programmtitel "Core" steht übrigens für "Cybernetic Organism Recovery Expedition". Laut Anleitung soll die Station aus über 1000 Räumen bestehen, die ich allerdings nicht nachgezählt habe. Auch hier sollte man wieder eine Karte zeichnen, um den Überblick nicht zu verlieren. Das Programm selbst ist grafisch gut gemacht, bietet jedoch kaum Überraschungen. Wie so oft muß man Teile untersuchen und aufnehmen und sich dabei natürlich vor zahlreichen Feinden in acht nehmen. Das Spiel ist dennoch guter Durchschnitt.

System Spectrum 48 K

Ralf Kappe

Schaltbild – CAD

Dieses Programm wendet sich an alle Hobby- und Amateurelektroniker, die ihre eigenen Schaltpläne erstellen. Will man selbst einen Schaltplan (z.B. eine Schnittstelle für den Computer) entwerfen, so geschieht dies zuerst in Gedanken und mit einigen Kritzeleien auf einem Blatt Papier. Das hier vorgestellte Programm erlaubt nun, die ersten Ideen für einen Schaltplan ordentlich aufzuzeichnen und auszudrucken.

Nach dem Start des Programms befindet man sich in einem Einstiegsmenü, welches z.B. die Änderung der Druckerart oder des Speichermediums ermöglicht. Bei Abspeicherung des Programms auf Microdrive werden automatisch alle erforderlichen Menus um die Microdrive-Optionen (Catalog, Erase etc.) erweitert, so daß man beim Laden von Cartridge sofort eine erweiterte Version vor sich hat.

Nach einem Sprung in die Zeichenroutine kann das Schaltbild aufgebaut werden. Dies geschieht mit Hilfe der Cursor- und verschiedener Funktionstasten. Da das Programm über eine sehr umfangreiche Bauteilesammlung verfügt (vom einfachen Widerstand bis zu verschiedenen ICs und logischen Gattern), sind eigentlich alle normalen Schaltungen möglich. Daneben ist noch wahlweise eine automatische oder manuelle Beschriftung der Bauelemente sowie das Einfügen von beliebigen Texten in die Schaltung erlaubt. Bei der automatischen Beschriftung werden alle Bauteile gleicher Art gezählt und im Schaltplan mit entsprechender Bezeichnung versehen (z.B. R1, R2, ...). Alle Texte im Schaltplan werden mit 64 Zeichen je Zeile geschrieben.

Nach Verlassen des Zeichen-Editors läßt sich noch jedes erfaßte Bauteil mit einem Wert versehen, woraus sich eine Liste der Bauteile bildet. Diese kann dann zusammen mit dem Schaltplan auf einem Drucker (normale oder 4fache Größe) ausgegeben werden und ergibt so eine vollständige Dokumentation der Schaltung. Als Ergänzung ist noch ein ganzseitiger Text-Editor eingebaut, um weitere technische oder sonstige Daten zur Schaltung niederschreiben zu können.

Natürlich lassen sich alle erstellten Schaltpläne auf das aktuelle Speichermedium retten, um sie später einmal zu ändern oder zu erweitern.

Da das Programm zum großen Teil in Basic geschrieben wurde, ist schnelles Zeichnen zwar nicht so ohne weiteres möglich, aber die eigentliche Entwicklungsarbeit findet ja sowieso im Kopf statt.

Weitere Informationen:
A. Thiesen
Bad[illegible]straße 70
5410 Höhr-Grenzhausen

Thomas Schröder

Brandneues Special

von

elektor

Interface-Elektronik

Auf dieses Heft haben die Computer-Freaks schon lange gewartet! Mit dieser Zeitschrift haben Sie die Möglichkeit sich Ihre entsprechende Schnittstelle selbst zu bauen.

Elektor Plus 5
INTERFACE-ELEKTRONIK

Hier ein kleiner Themenauszug:

Grundlagen
Info-Artikel über Sinn, Zweck und Anwendungsmöglichkeiten von Interfaces, z.B. zur Steuerung von Robotern oder das Messen von Lautsprecherboxen

Marktberichte
Übersicht von industriell hergestellten Interfaces und Zubehör

Bauanleitungen
* verschiedene Computer und Peripheriegeräte sind anschließbar

* RS232/Centronics Schnittstelle
* RS232 Analysator
* C64/128 Schnittstelle
* C64 Kassetteninterface
* MSX-Erweiterungen
* I/O-Erweiterung für MSX Computer
* Modem mit Selbstwahl-Interface
* Universeller I/O-BUS
* I/O-BUS-Karte Serielle Schnittstelle
* I/O-BUS-Karte Kanalwähler
* I/O-BUS-Karte Analogeingang
* I/O-BUS-Karte Echtzeituhr

Schauen Sie mal rein, es lohnt sich!

Erscheinungstermin ist am
21 November 1986

Direkt beim Verlag oder im Fachhandel
erhältlich DM 15,—

Elektor Verlag, Süsterfeldstr. 25, 5100 Aachen

Window

Mit der Window-Routine lassen sich beliebige rechteckige Bildschirmausschnitte manipulieren: SCROLL und ROLL in alle 4 Richtungen, CLS, Window abspeichern und wieder in den Bildschirmspeicher zurückladen (auch an andere Positionen). Attribute werden – da das Programm die einzelnen Optionen pixelweise ausführt – nicht verändert. Die Handhabung der Window-Routine ist etwas kompliziert, denn es müssen 6 Parameter übergeben werden. Der Aufruf erfolgt mit USR a: PRINT b, c, d, e, f, g. Die Variablen haben folgende Bedeutung:

a Gibt die Adresse des Maschinencodes an, der trotz seiner fast 500 Byte frei im Speicher verschiebbar ist.

b und c Sie geben die obere linke Ecke des Windows an. Hierbei darf b Werte von 0-191 und c von 0-31 besitzen. Daraus ergibt sich, daß in der Breite immer nur ganze Print-Positionen abgespeichert werden können.

d Bestimmt die Höhe des Windows und darf im Bereich von 1-192 liegen.

e Gibt die Breite des Windows an und kann Werte von 1-32 annehmen.

f Gibt die Adresse an, ab der das Window abgespeichert bzw. abgerufen werden soll. Der Wert ist außer bei den Optionen "Abspeichern" und "Zurückladen" überflüssig, muß jedoch immer mit angegeben werden.

g Sie gibt als letzte und wohl wichtigste Variable an, welche Funktion ausgeführt werden soll. Folgende Werte ergeben nachstehende Funktionen:

0 Scroll links
1 Roll links
2 Scroll rechts
3 Roll rechts
4 Scroll hoch
5 Roll hoch
12 Scroll runter
13 Roll runter
16 Window im Speicher ablegen
48 Window aus Speicher holen
128 Window CLS

Alle Scroll/Roll-Operationen werden pixelweise ausgeführt. Bei falschen Werten (z.B. Window zu groß) wird die Fehlermeldung A ausgegeben. Abgespeicherte Windows kann man durch Ändern der x- und y-Position leicht duplizieren, wobei sich schöne Effekte erzielen lassen. Beim Abspeichern der Windows sollte man darauf achten, daß man keine Daten zerstört. Es können allerdings nicht viele große Screens abgespeichert werden, da der ganze Spectrum-Screen (ohne Attribute) schon 6144 Byte lang ist. Der Speicherplatzbedarf eines Windows errechnet sich aus Breite×Höhe, so läßt sich leicht der Anfang des nächsten Windows bestimmen. Anstelle von festen Integerwerten kann man für b, c, d, e, f und g auch Variablen u.a. einsetzen, was den Umgang mit der Routine wesentlich erleichtert.

Dirk Zwers

MC-Generator

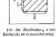
```
   1 INPUT "STARTADRESSE DES MCO
DES ? ";MCODE
   2 LET A=MCODE
   3 FOR B=1 TO 50
   4 RESTORE (B+19)
   5 LET C=0
   6 FOR D=1 TO 10
   7 READ E
   8 LET C=C+E
   9 POKE A,E
  10 LET A=A+1
  11 NEXT D
  12 READ F
  13 IF F<>C THEN  FLASH 1: PRIN
T ''"FEHLER IN DATAZEILE ";B+19'
'"GELESENE PRUEFSUMME: ";F'"ERRE
CHNETE PRUEFSUMME: ";C: FLASH 0:
 BEEP .1,-35: STOP
  14 PRINT "DATAZEILE ";B+19;" I
ST O.K."
  15 NEXT B
  16 CLS : PRINT  BRIGHT 1;"
 ALLE DATAS SIND O.K.         "
  17 PRINT  BRIGHT 1;" MIT  GOTO
 9990  KANN DAS PRO-    GRAMM UND
 DER ERZEUGTE CODE AB-"
  18 PRINT  BRIGHT 1;"        GESP
EICHERT WERDEN.        "
  19 PRINT  BRIGHT 1;"   BEI  CO
NTINUE  ERFOLGT EIN
  KURZDEMO.            ": STOP
  20 DATA 243,223,254,58,192,231
,22,6,213,231,1673
  21 DATA 205,130,20,205,153,30,
209,197,21,32,1210
  22 DATA 243,193,121,254,7,56,2
0,254,12,48,1208
  23 DATA 16,254,13,48,12,254,16
,48,8,254,907
  24 DATA 48,48,4,254,128,32,26,
231,33,176,962
  25 DATA 92,221,113,0,253,225,2
09,123,209,83,1528
  26 DATA 95,193,121,193,65,79,1
23,167,48,3,1070
  27 DATA 120,48,7,253,33,58,92,
251,207,9,1006
  28 DATA 254,193,48,245,123,167
,48,241,129,56,1496
  29 DATA 238,254,23,48,234,221,
203,6,102,32,1365
  30 DATA 119,221,203,6,94,48,4,
120,130,71,1002
  31 DATA 5,197,213,120,230,7,87
,120,230,248,1457
  32 DATA 15,15,15,71,15,15,15,2
30,224,129,744
  33 DATA 111,120,230,24,246,64,
130,103,209,213,1450
  34 DATA 221,203,0,126,40,14,54
,0,35,29,722
  35 DATA 32,250,209,193,4,21,32
,209,251,201,1402
  36 DATA 22,0,221,203,0,86,32,5
0,221,203,1030
```

```
  37 DATA 0,78,40,48,125,131,61,
111,203,62,859
  38 DATA 245,29,40,12,43,203,62
,48,4,35,721
  39 DATA 203,254,43,29,32,244,2
21,203,0,70,1299

  40 DATA 40,7,241,48,5,203,254,
24,1,241,1064
  41 DATA 209,193,4,21,32,151,25
1,201,24,105,1191
  42 DATA 24,31,203,38,245,29,40
,12,35,203,860

  43 DATA 38,48,4,43,203,198,35,
29,32,244,874
  44 DATA 221,203,0,70,40,219,24
1,48,217,203,1462
  45 DATA 198,24,213,221,203,0,2
14,197,213,120,1603

  46 DATA 230,7,87,120,230,240,1
5,15,15,71,1030
  47 DATA 15,15,15,230,224,129,1
11,120,230,24,1113
  48 DATA 246,64,130,103,209,213
,221,203,0,110,1499

  49 DATA 40,6,229,253,229,225,2
53,225,126,253,1839
  50 DATA 119,0,35,253,35,29,32,
246,209,193,1151

  51 DATA 221,203,0,110,40,3,229
,253,225,4,1288
  52 DATA 21,32,198,251,201,209,
193,217,0,245,1567
  53 DATA 197,213,229,217,0,21,4
8,80,197,213,1415

  54 DATA 229,217,225,209,193,21
7,213,22,0,25,1550
  55 DATA 43,209,126,245,43,29,3
2,250,217,213,1407
  56 DATA 217,209,197,213,8,62,2
,8,120,230,1266

  57 DATA 7,87,120,230,248,15,15
,15,71,15,823
  58 DATA 15,15,230,224,129,111,
120,230,24,246,1344
  59 DATA 64,130,103,209,193,197
,213,4,8,61,1182

  60 DATA 245,0,241,48,15,229,25
3,225,221,203,1680
  61 DATA 0,94,40,2,5,5,24,206,2
4,67,467
  62 DATA 126,253,119,0,253,35,3
5,29,32,246,1128
  63 DATA 209,193,221,203,0,94,4
0,2,5,5,972
  64 DATA 4,21,32,174,213,120,25
0,7,87,120,1006
  65 DATA 230,246,15,15,15,71,15
,15,15,230,867
  66 DATA 224,129,111,120,230,24
,246,64,130,103,1381
  67 DATA 209,241,221,203,0,70,4
0,3,119,24,1130
  68 DATA 2,54,0,35,29,32,240,25
1,217,0,860
  69 DATA 225,209,193,241,217,0,
251,201,8,0,1545
8999 REM    WINDOW DEMO
9000 CLS : LIST 9000
9010 INPUT "X-POS. DES WINDOWS ?
 ";A
9020 INPUT "Y-POS. DES NINDOWS ?
 ";B
9030 INPUT "HOEHE DES WINDOWS ?
";C
9040 INPUT "BREITE DES WINDOWS ?
 ";D
9050 INPUT "ADRESSE FUER EVENTUE
LLEN ZUGRIFFEINES 2. WINDOWS ? "
;E
9060 INPUT "FUNKTION ? ";F
9070 PRINT #0;AT 1,0;"BITTE DRUE
CKEN SIE EINE TASTE ''": PAUSE 0
9080 FOR G=1 TO 100: LET Z=USR M
CODE: PRINT A,B,C,D,E,F: NEXT G
9090 PRINT #0;AT 1,0;"BITTE DRUE
CKEN SIE EINE TASTE ''": PAUSE 0
9990 LET A2=INT (MCODE/256): LET
 A1=MCODE-256*A2
9991 POKE 23728,A1: POKE 23729,A
2
9992 CLEAR
9993 SAVE "WINDOW" LINE 1: SAVE
"WINDOWCODE"CODE (PEEK 23728+256
*PEEK 23729),500
9994 PRINT  FLASH 1;AT 10,0;"
        VERIFY '        "
9995 VERIFY "WINDOW": VERIFY "WI
NDOWCODE"CODE (PEEK 23728+256*PE
EK 23729),500
9996 RUN
9999 REM ▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚
         ▚ 1986 BY G-S-S-T    ▚
         ▚ 1986 BY DIRK ZWEERS▚
         ▚       DAUNERSTR. 15▚
         ▚       5000 KOELN 41▚
         ▚       0221/448769  ▚
         ▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚
```

Respos für ZX 81

Für die kleine Zahl unermüdlicher Fans des immer noch leistungsstarken ZX 81 ist es wichtig, weiterhin mit Neuerungen versehen zu werden.

"Respos" ist ein Maschinenprogramm, das viele Routinen zur Bildschirmbearbeitung enthält. Dabei geht es um die drei Grundfunktionen Füllen (FIL), Löschen (CLR) und Invertieren (INV). Ziel des Programms ist es, beliebige Rechtecke blitzschnell zu bearbeiten. Es bieten sich zwei Lösungen an. Die reihenweise Bearbeitung (RILF=reihenweise invertieren löschen füllen) und die Bearbeitung durch eine im Rechteck umlaufende Spirale (SILF=spiralig invertieren löschen füllen). Die letzte Methode hat den entscheidenden Vorteil, daß man die Spirale nach beliebig vielen Umläufen stoppen kann (Erzeugung von Rahmen). Auch ist im Hinblick auf optische Effekte ihre Geschwindigkeit steuerbar. Aus den beiden Grund-Routinen RILF und SILF setzen sich weitere zusammen. So gibt es abgekürzte Befehle für den ganzen Bildschirm und für ein symmetrisch zum Bildschirm liegendes Rechteck (Innenfeld ohne Rahmen). Schließlich erhält man noch 7 Effekte, die sich vor allem beim Invertieren verblüffend ausmachen.



Für die Bearbeitung eines Rechtecks ist es zunächst nötig, 4 Koordinaten für die Lage des Rechtecks einzugeben. "Respos" enthält zu diesem Zweck eine spezielle Routine, die es gestattet, alle erforderlichen Eingaben und den Startbefehl der gewünschten Routine in einer Basic-Zeile zu erledigen.

Rechteckkoordinaten

4 Zahlen sind in folgender Reihenfolge einzugeben.

A = Zahl der freien Zeilen über dem Rechteck
B = Zahl der freien Spalten links vom Rechteck
C = Zahl der Rechteckzeilen = Höhe des Rechtecks
D = Zahl der Rechteckspalten = Breite des Rechtecks

Befehle auf der Basis von RILF

Der Grundbefehl lautet PRINT USR RILF,A,B,C,D,ART.
Für A, B, C, D werden die erwähnten Rechteckkoordinaten eingegeben, für die Variable ART entweder CLR (löschen), INV (invertieren) oder FIL,X, wobei X der Code des zu printenden Zeichens ist. Für die Bearbeitung des gesamten Bildschirms gibt es RILF-TOTAL. Hier entfällt die Eingabe der Rechteckkoordinaten, so daß der reduzierte Befehl folgendermaßen lautet: PRINT USR RTOT,ART.
Schließlich kann auch ein Innenfeld ohne Rand, also ein zum Bildschirm symmetrisch gelegenes Rechteck bearbeitet werden. Der Kurzbefehl für RILF-INNEN lautet: PRINT USR RIN,Z,ART.
Hier gibt Z die Zahl der freien Zeilen um das Rechteck (Innenfeld) an. ART kann durch CLR, INV oder FIL,X ersetzt werden.

Befehle auf der Basis von SILF

Der Grundbefehl lautet hier PRINT USR SILF,A,B,C,D,ART,BORDER,SLOW.
Wie oben stellen A bis D die Rechteckkoordinaten dar. ART kann CLR, INV oder FIL,X sein. BORDER gibt die Zahl der auszuführenden Spiralumläufe an. BORDER=0 bedeutet z.B., daß das Rechteck ganz von der Spirale ausgefüllt wird. SLOW steuert die Geschwindigkeit der umlaufenden Spirale. 0 gibt den schnellsten Umlauf an, mit zunehmendem SLOW-Wert wird das Tempo langsamer. Beispiel: PRINT USR SILF,2,5,10,12,INV,2,3 printet eine Rechteckspirale mit 2 Umläufen im Tempo 3 in Form einer Inversion der schon vorhandenen Zeichen.

PRINT USR STOT,ART,BORDER,SLOW ist die abgekürzte Version für SILF über den ganzen Bildschirm und PRINT USR SIN,Z,ART,BORDER,SLOW die Version für ein Innenfeld ohne Rand. Beispiel: Die Befehlsfolge PRINT USR STOT,FIL,23,3,0: PRINT USR SIN,2,INV,1,0 und PRINT USR RIN,3,FIL,8 erzeugt im höchsten Tempo (SLOW=0) einen 3fachen Rahmen mit Sternchen um den ganzen Bildschirm, invertiert die innerste Zeile und füllt das Innenfeld grau aus.

Aus diesen 6 Grundbefehlen lassen sich beliebige eigene Effekte kombinieren, wie das Demo-Programm zeigt. Das Programm "Respos" enthält jedoch solche Effekt-Kombinationen im Maschinencode. Ihre Adressen befinden sich in einer Liste am Programmende. Damit wurde erreicht, daß der Aufruf allein durch Angabe der Effekt-Nummer möglich ist: PRINT USR EFF,NR,ART,SLOW.

Wie man erkennt, ist NR die erste einzugebende Zahl, dann die ART der Bearbeitung und schließlich das Tempo. Die 7 Effekte (NR = 1 bis 7) sind am wirkungsvollsten bei der Inversion, machen sich aber auch beim Löschen des gesamten Bildschirms gut. Wer will, kann auch FIL,X versuchen, hier ist SLOW = 5 ein durchaus interessanter Wert. Folgende Möglichkeiten stehen bei den Effekt-Kombinationen zur Verfügung:

EFF 1	Zeilenablauf von oben nach unten
EFF 2	Spaltenablauf von links nach rechts
EFF 3	Kontraktion horizontal
EFF 4	Kontraktion vertikal
EFF 5	Kontraktion horizontal und vertikal
EFF 6	Konzentrisch wachsende Rechtecke
EFF 7	Spirale von innen nach außen

Vorsichtsmaßnahmen

"Respos" enthält keine Absicherung gegen falsche Eingaben. Folglich müssen einige Dinge beachtet werden, will man nicht das gesamte Programm zum Absturz bringen. Die Summe der Rechteckkoordinaten A+C darf den Wert 22 nicht übersteigen, ebenso gilt B+D < 33. Für Innenfeldbefehle RIN,Z und SIN,Z muß Z < 11 sein. Keine Eingabe darf fehlen oder zuviel vorhanden sein, denn die spezielle Routine erwartet die genau vorgeschriebene Zahl von Eingaben. Ferner darf der Code zu printender (FIL) Zeichen nur aus 0 bis 63 und 128 bis 191 gewählt werden.

Eingabe des Programms

Das Programm benötigt eine REM-Zeile mit 570 Byte (beliebige Zeichen). Diese erzeugt man mittels REM-Füller (CK 2/86) oder indem man 1 REM mit 186 Zeichen eintippt (5 Zeilen ganz, die 6. mit 26 Zeichen) und anschließend durch Editieren und Umnumerieren 2 REM und 3 REM erzeugt. Nun folgen die Befehle POKE 16510,0: POKE 16511,60: POKE 16512,2: POKE 16514,0 bis und POKE 16515,118. Durch New Line ist jetzt nur noch die Zeile 0 REM zu erkennen. Wer einen REM-Füller benutzt, sollte auf jeden Fall auch POKE 16510,0 als Editierschutz eingeben.

Danach tippt man Listing 1 ab und startet mit RUN. Sollte ein falscher Wert enthalten sein, meldet der Computer über die eingebauten Checksummen, in welchem Statement der Fehler sitzt. Ein weiterer Hinweis ist nützlich, da das Listing immer wieder auf 0 REM zurückspringen wird. Hier bringt z.B. die Eingabe von LIST 10 gefolgt von POKE 16420,100 Abhilfe. Damit stabilisiert sich das Bild.

Initialisierung

Nach Eingabe des Listings und Einpoken des Maschinenprogramms durch RUN wird das Listing ab 10 nicht mehr benötigt und sollte gelöscht werden. Dafür benötigt man folgende Initialisierung, damit die 7 Basic-Befehle von "Respos" wirksam werden können.

```
10 LET INV=145
20 LET CLR=150
30 LET FIL=154
40 LET RILF=16965
50 LET SILF=16976
60 LET RTOT=16987
70 LET STOT=16996
80 LET RIN=17016
90 LET SIN=17031
100 LET EFF=17049
```

Demo-Programm

Das Demo-Programm setzt „RESPOS" in 0 REM voraus, enthält die Initialisierung und zeigt die Wirkung der eingebauten Routinen sowie Möglichkeiten, durch geeignete Programmierung neue Effekte zu erzielen. Innerhalb des Programms wird an verschiedenen Stellen darauf hingewiesen, welcher "Respos"-Befehl angewandt wird bzw. ab welcher Programm-Nummer die gezeigte Routine läuft, so daß man schnellen Zugang zu eigenen Anwendung findet.

Friedrich W. Buckel

Gegen Einsendung von 10,- DM an F. Buckel, Schulstr. 15, 7184 Kirchberg/Jagst erhalten Sie ein kommentiertes Assembler-Listing sowie „Respos" und das Demo-Programm auf Cassette.

Listing 1

```
EINGABE-PROGRAMM ZU RESPOS
--------------------------------
NACH ERZEUGUNG DER ZEILE  0 REM

  10■REM EINGABE ZU RESPOS
         F. W. BUCKEL
         7184 KIRCHBERG/JAGST
  15 FAST
  20 LET A=16514
  30 LET N=0
  40 LET T$="767600000201101011
87AB320F6C97EC68077C9AF32584036
0C92A0C40"
  50 LET CS=2656
  60 GOSUB 700
  70 LET T$="2D458440C5AFB928071
12100411910FDC1AFE8920032310FDED5
B8640C9CD"
  80 LET CS=3352
  90 GOSUB 700
 100 LET T$="8840CD9840C9CD0D40A
F66C08AC8434A3E21915F1600C541230
0C0010FA"
 110 LET CS=3710
 120 GOSUB 700
 130 LET T$="10C110F4C900C09D403
AE140473A0640A700003032844010421
8AF68C0D5"
 140 LET CS=3429
 150 GOSUB 700
 160 LET T$="23CD8D4010FA01431DA
F66C0D511210019CD8D4010F7014215A
F66C0D528"
 170 LET CS=3743
 180 GOSUB 700
 190 LET T$="CD8D4010FA01431DAFB
8C805112100A7ED52CD8D4010F5D13A6
4408BC818"
 200 LET CS=3930
 210 GOSUB 700
 220 LET T$="6D2A0C401E151620C9A
F3265403E20328740CDC440C9AF32544
03E163286"
 230 LET CS=2734
 240 GOSUB 700
 250 LET T$="4010F121064407E23774
F3E16910128 77C60A2377C9CD5741160
AC05741CD"
 260 LET CS=3169
 270 GOSUB 700
 280 LET T$="E240C90606753D39844
0C5CD5741CDC440CD8840C110EEC9060
6783D3284"
 290 LET CS=3367
 300 GOSUB 700
 310 LET T$="40C5CD57413E0132E14
0CDE240C110ECC906203E01328740C5C
53E209032"
 320 LET CS=3193
 330 GOSUB 700
 340 LET T$="8840CD4C41F13D32854
0CD4C41C1CD984010E7C906163E01328
640C5C83E"
 350 LET CS=3279
 360 GOSUB 700
 370 LET T$="1690328440CD3F41F13
D328440CD3F41CD3840C110E7C9CDA14
118DA210A"
 380 LET CS=3896
 390 GOSUB 700
 400 LET T$="41E5386ECDC241E1383
FC9216941E536C1CDA141E1364CC90FE
73A250FFE"
 410 LET CS=4140
 420 GOSUB 700
 430 LET T$="FDF8CC620FF1C4550FC
DA70EC90604218440C5E6CD0242E1718
3C110F3CD"
 440 LET CS=3893
 450 GOSUB 700
 460 LET T$="024221C140713E0A508
007CD024221984371C0C0024221E1407
1CD024221"
 470 LET CS=2784
 480 GOSUB 700
 490 LET T$="8A4071C9CD1542CDC44
0CFFFCD1542CD3942CDE240CFFFCD254
2CD3741CD"
 500 LET CS=4150
 510 GOSUB 700
 520 LET T$="C740CFFFCD2549CD394
2CD37417E3285640CDE540CFFFCD0242
1544071CD"
 530 LET CS=3637
 540 GOSUB 700
 550 LET T$="2542CD6941CFFFCD024
22184407 1CD2542CD3942CD6E41CFFFC
D0242D600"
 560 LET CS=3312
 570 GOSUB 700
 580 LET T$="218142097E21AE4877C
D2542CD4042CD6A41CFFFEBF0A1C2E37
53A000000"
 590 LET CS=3530
 600 GOSUB 700
 610 PRINT "RESPOS IST FERTIG,ZE
ILEN 10 BIS 700 LOESCHEN"
 620 SLOW
 630 STOP
 700 LET S=0
 710 FOR I=1 TO 60 STEP 2
 720 POKE A,16*CODE T$(I)+CODE T
$(I+1)-476
 730 LET S=S+PEEK A
 740 LET A=A+1
 750 NEXT I
 760 LET N=N+1
 770 IF S=CS THEN RETURN
 780 PRINT "FEHLER IN STATEMENT "
,10+N*30
```

Listing 2

```
   0 REM (ENTHAELT RESPOS)
  10■REM DEMO ZU RESPOS (C) 5/86
BY       F.U.BUCKEL SCHULSTR.15
         7184 KIRCHBERG/JAGST
  20 LET INU=145
  30 LET CLA=150
  40 LET FIL=154
  50 LET RILF=16988
  60 LET SILF=16976
  70 LET RTOT=16957
  80 LET STOT=16998
  90 LET RIN=17016
 100 LET SIN=17031
 110 LET EFF=17046
 120 LET PAUSE=200
 130 LET TASTE=230
 140 GOTO 500
 200 FOR I=1 TO 50
 210 NEXT I
 220 RETURN
 230 PRINT AT 19,23,T$
 240 IF INKEY$="" THEN GOTO 240
 250 PRINT AT 19,23,L$( TO 8)
 260 RETURN
```

```
270 GOSUB TASTE
280 CLS
290 RETURN
300 DIM L$(32)
310 DIM Z(8)
311 LET Z(1)=128
312 LET Z(2)=160
313 LET Z(3)=8
314 LET Z(4)=8
315 LET Z(5)=61
316 LET Z(6)=23
317 LET Z(7)=20
318 LET Z(8)=27
320 LET T$="[illegible]"
330 RETURN
500 POKE 16418,0
510 PRINT USR STOT,FIL,23,5,0
520 PRINT USR RIN,6,FIL,8
530 PRINT AT 7,8,"DEMO-PROGRAMM
ZU",AT 11,10,"R E S P O S",AT 1
4,8,"VON F. W. BUCKEL"
540 PRINT USR SIN 3,INU,0,3
550 GOSUB PAUSE
560 PRINT USR RTOT,INU
570 GOSUB 320
580 GOSUB PAUSE
590 PRINT USR RIN,4,CLR
600 PRINT AT 6,8,"█",AT 8,8,"█
█",AT 10,8,"█",AT 12,8,"█"
610 GOSUB PAUSE
620 PRINT AT 6,10,"IHENUEISES",
AT 8,10,"IRRLARTIGES",AT 10,9,"R
GRNISATIONS-",AT 12,9,"YSTEM"
630 GOSUB PAUSE
640 PRINT AT 14,6,"7 NEUE BASIC
-BEFEHLE",AT 16,6,"ZUR BILDGESTA
TUNG"
650 GOSUB PAUSE
660 PRINT USR EFF 7,INU,1
670 PRINT USR STOT,CLR,0,1
680 PRINT AT 0,0,"SO PRINTET DE
R ZX-81 'EIN RECHTECK"
690 FOR I=1 TO 12
700 PRINT AT 4+I,8,"[illegible]
[illegible]"
710 NEXT I
720 PRINT AT 20,0,"UND SO MIT █
E S P O S"
730 GOSUB 270
740 PRINT AT 21,0,"[illegible]",
,"1.BEFEHL",,"PRINT USR RILF A,
B,C,D,FIL,49","USR RILF,5,8,12 18
,FIL,49"
750 GOSUB PAUSE
760 PRINT AT 2,0,"1",TAB 0,"2",
TAB 0,"3",TAB 0,"4",TAB 0,"[illegible]..
..."
770 GOSUB PAUSE
780 PRINT AT 0,0,"1234567[illegible]",T
AB 7,".",TAB 7,".",TAB 7,".",TAB
7,"."
790 GOSUB PAUSE
800 FOR I=1 TO 9
810 PRINT AT 4+I,8,CHR$ (156+I)
820 NEXT I
830 PRINT AT 14,0,"█",TAB 8,"█"
,TAB 5,"[illegible]"
840 GOSUB PAUSE
850 FOR I=1 TO 9
860 PRINT AT 6,8,"123456789[illegible]
[illegible]",AT 4,26,"[illegible]"
870 PRINT AT 17,15,"COD L=49"
880 GOSUB TASTE
890 PRINT USR EFF 3,CLR,6
900 PRINT AT 23,0,L$,AT 23,0,"W
EITERE BEISPIELE"
910 PRINT USR RILF,0,0,10,10,FI
L,8
920 PRINT USR RILF,6,16,14,21,F
IL,130
930 GOSUB PAUSE
940 PRINT USR RILF,4,4,12,12,IN
U
950 GOSUB PAUSE
960 PRINT USR RILF 0,0,1,31,FIL
,51
970 PRINT USR RILF,0,20,21,1,FI
L,51
980 GOSUB PAUSE
990 GOSUB PAUSE
1000 PRINT USR EFF,2,CLR,10
1010 FOR I=1 TO 11
1020 PRINT USR RILF,I,I,11,21,FI
L,I+35
1030 NEXT I
1040 PRINT USR RILF 12,12,9,19,C
LR
1050 GOSUB PAUSE
1060 PRINT USR RILF,12,12,9,19,I
NU
1070 GOSUB PAUSE
1080 PRINT AT 22,0,"[illegible]","P
RINT USR ,SILF A,B C,D,ART,BORD
ER,SLOW"
1090 PRINT USR EFF,1,CLR,7
1100 GOSUB PAUSE
1110 PRINT USR SILF,2,5,10,12,FI
L,180,1,0
1120 PRINT USR SILF,13,1,8,30,FI
L,81,2,0
1130 FOR I=1 TO 7
1140 PRINT USR SILF,13,1,8,30,IN
U,1,INT (4-I/2)
1150 NEXT I
1160 GOSUB TASTE
1170 PRINT AT 22,0,L$,L$,AT 22,0
,"3.BEFEHL",TAB 0,"PRINT USR RT
OT,INU"
1175 GOSUB TASTE
1180 PRINT USR RTOT,INU
1190 GOSUB TASTE
1200 PRINT AT 22,0,"[illegible]",TAB 0,"[illegible]
[illegible]"
1205 GOSUB TASTE
1210 PRINT USR STOT,FIL,21,4,0
1220 GOSUB TASTE
1230 PRINT AT 23,0,L$,AT 22,0,"5
.BEFEHL",,"PRINT USR RIN 4,FIL,
5"
1240 GOSUB TASTE
1250 PRINT USR RIN,4,FIL,8
1260 GOSUB TASTE
1270 PRINT AT 23,0,L$,AT 22,0,"6
.BEFEHL",,"[illegible],ART,
BORDER,SLOW"
1280 GOSUB TASTE
1290 FOR I=0 TO 7
1300 PRINT USR SIN,I,FIL,Z(I+1),
1,0
1310 NEXT I
1320 GOSUB PAUSE
1330 PRINT AT 22,0,L$,L$,AT 23,0
,"PRINT USR RIN,8,FIL,X"
1340 GOSUB PAUSE
1350 FOR X=0 TO 63
1360 PRINT USR RIN,8,FIL,X
1370 NEXT X
1380 FOR X=191 TO 128 STEP -1
1390 PRINT USR RIN,8,FIL,X
1400 NEXT X
1410 GOSUB PAUSE
1420 PRINT AT 22,0,L$,L$,AT 22,0
,"7.BEFEHL",,"PRINT USR EFF [illegible],IN
ART,ANZ"
1430 GOSUB PAUSE
1440 FOR I=1 TO 5
1450 PRINT USR EFF,5,INU,1
1460 NEXT I
1470 PRINT USR EFF,1,INU,7
1480 PRINT USR EFF,1,INU,7
1490 PRINT USR EFF,2,INU,4
1500 PRINT USR EFF,2,INU,4
1600 PRINT USR EFF,6,CLR,8
1610 PRINT AT 22,0,L$,L$
1620 PRINT USR STOT,FIL,23,4,0
1630 PRINT AT 7,6,"ANWENDUNGSBEI
SPIELE",AT 10,15,"ZU",AT 13,11,
"R E S P O S"
1640 GOSUB PAUSE
1650 FOR I=1 TO 11
1660 PRINT USR RILF,0,0,I,5+I,IN
```

```
U
1870 PRINT USR RILF,22-I,0 I,5+I
,INV
1880 PRINT USR RILF,0,27-I,I,5+I
,INV
1890 PRINT USR RILF,22-I,27-I,I,
5+I,INV
1700 NEXT I
1710 PRINT AT 23,0,"GEKOPPELTE R
ILF-ROUTINEN AB 1550"
1720 GOSUB TASTE
1730 PRINT USR STOT,CLR
1740 PRINT USR STOT,INV
1750 PRINT AT 23,0,L$;AT 23,0,"T
ABELLEN-GRAFIK MIT RILF AB 1730"
1760 PRINT USR RILF,1,1,9,14,CLR
1770 PRINT USR RILF,1,16,4,15,FI
L,138
1780 PRINT USR RILF,6,16,4,15,CL
R
1790 PRINT USR RILF,11,7,10,24,C
LR
1800 PRINT USR RILF,11,1,10,5,FI
L,5
1810 GOSUB TASTE
1820 PRINT USR STOT,CLR,0,1
1830 PRINT AT 23,26,"[illegible]"
1840 PRINT USR RILF,3,0,1,31,INV
1850 FOR I=4 TO 31 STEP 7
1860 PRINT USR RILF,0,I,21,1,FIL
,128
1870 NEXT I
1880 GOSUB TASTE
1890 PRINT USR EFF,4,CLR,5
1900 PRINT AT 23,26,"1910"
1910 FOR I=0 TO 12 STEP 6
1920 PRINT USR RILF,0,I,21,1,FIL
,128
1930 NEXT I
1940 FOR I=0 TO 20 STEP 5
1950 PRINT USR RILF,I,0,1,19,FIL
,128
1960 NEXT I
1970 GOSUB PAUSE
1980 FOR I=1 TO 19
1990 PRINT USR RILF,0,0,21,19,IN
V
2000 NEXT I
2010 GOSUB PAUSE
2020 PRINT AT 23,0,L$;AT 23,0,"I
NVERSIONSEFFEKTE AB 2020"
2030 PRINT USR STOT,FIL,6,0,1
2040 PRINT USR STOT,INV,0,1
2050 PRINT USR STOT,INV,2,0
2060 PRINT USR EFF,1,INV,2
2070 PRINT USR EFF,2,INV,5
2080 PRINT USR EFF,5,INV,3
2090 PRINT USR EFF,7,INV,0
2100 FOR I=1 TO 15
2110 PRINT USR STOT,INV,0,0
2120 NEXT I
2130 PRINT USR EFF,6,INV,6
2140 PRINT USR EFF,0,CLR,10
2150 PRINT AT 22,0,L$,L$,AT 23,4
,"ZUFALLSPROGRAMM AB 2160",AT 10
,9,"COMPUTERKUNST",USR STOT,FIL,
135,2,0
2160 GOSUB PAUSE
2170 DIM C(20)
2180 LET C(1)=0
2190 LET C(2)=1
2200 LET C(3)=3
2210 LET C(4)=5
2220 LET C(5)=7
2230 LET C(6)=8
2240 LET C(7)=9
2250 LET C(8)=18
2260 LET C(9)=20
2270 LET C(10)=23
2280 LET C(11)=24
2290 LET C(12)=49
2300 LET C(13)=52
2400 LET C(14)=61
2410 LET C(15)=128
2420 LET C(16)=133
2430 LET C(17)=137
2440 LET C(18)=161
2450 LET C(19)=176
2460 LET C(20)=189
2470 PRINT USR RIN,8,CLR
2480 LET COD=C(INT (RND*19)+2)
2490 PRINT USR STOT,FIL,COD,2,0
2500 LET J=INT (RND*10)+4
2510 FOR I=1 TO J
2520 LET A=INT (RND*17)+3
2530 LET C=INT (RND*10)+2
2540 IF A+C>16 THEN GOTO 2520
2550 LET B=INT (RND*15)+3
2560 LET D=INT (RND*13)+2
2570 IF B+D>28 THEN GOTO 2550
2580 PRINT USR RILF,A,B,C,D,FIL,
C(INT (RND*20)+1)
2590 NEXT I
2600 GOSUB PAUSE
2610 GOSUB PAUSE
2620 GOTO 2470
```

Spindizzy

Zunächst gab es "Marble Madness" ein Programm das besonders in Spielhallen ein Riesenhit wurde. Dann kamen die Adaptionen und Kopien für Heimcomputer. Auch "Spindizzy" fällt in diese Kategorie ebenso wie z. B. "Gyroscope" und vielleicht "Quazatron".

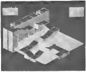
Das Programm zeigt einige Hundert aneinandergrenzende Ebenen, die eine fremde Welt darstellen sollen. Der Spieler muß mit einem Spezialgerät, das wie ein Kreisel aussieht, zur Erkundung gehen und wertvolle Edelsteine einsammeln. Leider kann das Gerät nur auf den vorgegebenen quadratischen Flächen bewegt werden. Weicht es davon ab, ist es verloren. Gerade hier in der Steuerung liegt das größte Problem für den Spieler, denn durch die vielen Kurven und Hindernisse muß er sich voll konzentrieren und lange üben.

Erschwerend kommt hinzu, daß kein Joystick verwendet werden kann, die Steuerung muß über die Tastatur erfolgen. Unerbittlich läuft auch noch die Zeit ab, die man aber durch je den gesammelten Edelstein wieder etwas verlängern kann. Zur Orientierung in diesem Labyrinth dient eine jederzeit abrufbereite Karte. Auch kann das Spiel eingefroren werden, wenn man eine Pause einlegen will.

"Spindizzy" mit seiner hervorragenden 3-D-Grafik und den Bewegungen des Kreisels ist den Programmierern wirklich gut gelungen. Man muß es sich einfach ansehen.

System Spectrum 48 K
Hersteller Electric Dreams

Stephan König


Spindizzy
Eine fremde Welt in 3-D-Grafik

Indexomat für den QL

Eine der vielen praktischen Eigenschaften des QL ist, daß er automatisch auf Cartridge oder Floppy ein Programm mit Namen „boot" sucht, es lädt und startet. Es wäre also nützlich, wenn nach Einlegen von Disk oder Cartridge ein durchnumeriertes Verzeichnis der Programme erschiene und man zum Laden nur noch die gewünschte Programmnummer eingeben müßte.

Dies ist die Aufgabe von „Indexomat". Da der QL über mehrere Lademöglichkeiten verfügt, wird das Programm auf Extensions untersucht; das sind drei Buchstaben, die dem Programm mittels Strich (wie beim Unterstreichen üblich) angefügt werden. Die Extensions entsprechen voll der von Psion Software und dem Sinclair Club Leverkusen e. V. empfohlenen Normung.

Nachfolgend eine Liste der verwendeten Extensions:

_bas für Basic-Programme
_exe für EXEC_W fähige Programme
_scr für abgespeicherte Bilder (Mode 8)
_cde für abgespeicherten Maschinencode (kann nicht vom Indexomat geladen werden)

Hinzu kommen die Extensions der Psion Software.

Wenn kein Extension gefunden wird, ruft man das Programm mit LRUN auf.

Bevor „Indexomat" verwendet werden kann, benötigt die Cartridge einen Header mit dem Namen inf100. Er wird mit folgenden Befehlen erzeugt:

OPEN_NEW #5, mdv1_inf100: DIR #5, mdv1_: CLOSE #5

Indexomat ist auch für Disketten vorgesehen; man muß nur device$="mdv1_" in device$="flp1_" ändern. Sollte auf der Cartridge oder Diskette nachträglich noch ein Programm eingefügt werden, so ist der Header nochmals zu erzeugen. Der Aufruf eines nicht vorhandenen Programms wird mit NO FILE quittiert.

Jürgen Malberg

Indexomat

```
5 MODE 4
10 device$="mdv1_"
15 DEFine PROCedure awindow:OPEN
#10,scr
20 WINDOW#10,465,205,30,10:CLS#1
0:PAPER#10,5:FOR scr=1 TO 100 ST
EP 2:BORDER#10,scr,2:NEXT scr:OP
EN#10,scr:WINDOW#10,465,205,30,1
0:BORDER#10,2,2:PAPER#10,4:INK#1
0,0:CLS#10
25 END DEFine
30 DEFine PROCedure rsx:CSIZE#0,
1,0:REPeat in:AT#0,1,0:INPUT#0,"
File ? ",nummer$:IF nummer$<>""
THEN nummer=nummer$+2: EXIT in
35 IF "_scr" INSTR file$(nummer)
 THEN MODE 8: LBYTES device$&fil
e$(nummer),131072:PAUSE:MODE 4:a
window:directory:rsx
40 IF "_bas"INSTR file$(nummer)
THEN LRUN device$&file$(nummer)
45 IF "_cde" INSTR file$(nummer)
 THEN CLS#0:CSIZE#0,2,1:PRINT#0,
"Can't load it":PAUSE:CSIZE#0,0,
0:CLS#0:rsx
50 IF "_exe" INSTR file$(nummer)
 THEN CLS: EXEC_W device$&file$(
nummer)
55 IF "_doc" INSTR file$(nummer)
 THEN CLS#0:CSIZE#0,2,1:PRINT#0,
"Can't load PSION dates. Use QUI
LL":PAUSE:CSIZE#0,0,0:CLS#0:rsx
60 IF "_aba" INSTR file$(nummer)
 THEN CLS#0:CSIZE#0,2,1:PRINT#0,
"Can't load PSION dates. Use ABA
CUS":PAUSE:CSIZE#0,0,0:CLS#0:rsx
65 IF "_grf" INSTR file$(nummer)
 THEN CLS#0:CSIZE#0,2,1:PRINT#0,
"Can't load PSION dates. Use EAS
EL":PAUSE:CSIZE#0,0,0:CLS#0:rsx
70 IF "_dbf" INSTR file$(nummer)
 THEN CLS#0:CSIZE#0,2,1:PRINT#0,
"Can't load PSION dates. Use ARC
HIVE":PAUSE:CSIZE#0,0,0:CLS#0:rs
x
75 IF "inf100"=file$(nummer) THE
N CLS#0:CSIZE#0,2,1:PRINT#0,"Hea
der info already read":PAUSE:CSI
ZE#0,0,0:CLS#0:rsx
80 IF LEN(file$(nummer))=0 THEN
CSIZE#0,2,1:PRINT#0,"No file":PA
USE:CSIZE#0,0,0:CLS#0:rsx
85 LRUN device$&file$(nummer):EN
D IF :END DEFine
90 END IF
95 DEFine PROCedure read_inf100
100 DIM file$(77,15)
105 OPEN_IN#5,device$&"inf100"
110 FOR info=0 TO EOF(#5)
115 FOR nui=1 TO 77
120 los=nui
125 INPUT#5,file$(nui)
130 IF EOF(#5) THEN LET nui=100:
END IF
135 END FOR nui:END FOR info:CLO
SE#5
140 END DEFine
145 DEFine PROCedure directory
150 CURSOR#10,100,1:CSIZE#10,1,0
:UNDER#10,1:PRINT#10,"Mediumname
="&file$(1)(1 TO 10):CSIZE#10,0,
0:CURSOR#10,0,25:UNDER#10,0
155 FOR filenum=3 TO 21
160 filenum2=filenum-2
165 AT#10,filenum2,1:PRINT#10,fi
lenum2&"."&file$(filenum)
170 END FOR filenum
175 FOR filenum=22 TO 40
180 filenum2=filenum-2
185 tab=filenum-21
190 AT#10,tab,19:PRINT#10,filenu
m2&"."&file$(filenum)
195 END FOR filenum
200 FOR filenum=41 TO 59
```

```
205 filenum2=filenum-2
210 tab=filenum-40
215 AT#10,tab,39:PRINT#10,filenu
m2&"."&file$(filenum)
220 END FOR filenum
225 FOR filenum=59 TO 77
230 filenum2=filenum-2
235 teb=filenum-58
240 AT#10,tab,58:PRINT#10,filenu
m2&"."&file$(filenum)
245 END FOR filenum
250 END DEFine
255 awindow:read_inf10:director
y:rex
```

Computergrafische Experimente mit Pascal

Von Karl-Heinz Becker
und Michael Dörfler
Vieweg-Verlag
320 Seiten, 42,-DM
ISBN 3-528-04461-6

"Apfelmännchen", "Fraktale", "Mandelbrot", "Feigenbäume", "Julia-Mengen" – dies alles sind Begriffe, die durch Computerzeitschriften und Computerclubs geistern. Fast jeder Computerfan hat ein "Apfelmännchen"-Programm, doch ist den wenigsten klar, daß das Ganze nicht bloß aus schönen, bizarren Grafiken besteht, sondern gestandene Mathematik und Gegenstand aktuellster Forschung in der Physik ist. Um so wichtiger ist ein Buch, das in die Materie einführt, die Sprache der Computerfans erläutert und zu eigenen Experimenten anregt.

Der Diplom-Informatiker Karl-Heinz Becker und der Diplom-Physiker Michael Dörfler, beide in der Lehrerfortbildung tätig, haben das vorliegende Buch über "Chaos und Ordnung in dynamischen Systemen" geschrieben. Dem ernsthaft interessierten Heimcomputerbesitzer ist das Buch sehr zu empfehlen. Es bringt all die Hintergrundinformationen, die man z. B. zum Verständnis des Programms "Mandelbrot" aus der CK benötigt.

Das Buch gliedert sich in drei Teile mit den Überschriften: Forscher entdecken das Chaos, Bausteine für grafische Experimente, Pascal und die Apfelmännchen. Den vierten Teil bildet der Anhang. Hier werden Computersystem und Pascal-Dialekt, mit denen gearbeitet wurde, erklärt. Dieser Teil enthält nützliche Tips zum Umschreiben der Programme in den eigenen Pascal-Dialekt.

Bei der Auswahl von Rechner (Apple) und Sprache (UCSD Pascal) haben die Autoren nicht sehr verbreitete Systeme ausgesucht. Aber zum Glück ist Pascal soweit standardisiert, daß die Anpassung an den eigenen Rechner keine Schwierigkeiten bereiten dürfte. Viele Programme sind auch in einer Art Metasprache formuliert, so daß der Algorithmus klar wird, das Ganze im Grunde jedoch von der Sprache unabhängig ist. Die Grafik-Routinen bedürfen aber in jedem Fall einer Anpassung, was nicht schwierig sein dürfte, da jeder Computerbesitzer ja wahrscheinlich die Grafik seines Rechners kennt.

Von den Programmen darf man allerdings keine hohen Geschwindigkeiten erwarten. Es liegt in der Natur dieser Probleme, daß sie sehr viel Rechenzeit benötigen, und der Spectrum ist schließlich kein Cray. Dennoch ist es erstaunlich, wie schnell dieser kleine Sinclair-Rechner mit dem Hisoft-Pascal rechnet. Ein Umsetzen der Programme in Basic sollten nur sehr geduldige Menschen vornehmen, da sich die Laufzeiten dann immer im Bereich von Stunden bewegen (Feigenbaum auf Spectrum ca. 35 Minuten in Basic, 1.5 Minuten in Pascal). Wer sich also mit dem Buch beschäftigen will, sollte die Anschaffung eines Pascal-Compilers nicht scheuen.

Das Angebot auf der dritten Umschlagseite sollte man ignorieren. Zehn Bilder aus dem Buch auf Diskette bzw. als Kopie eines Ausdrucks mit dem Laserdrucker für 200,-DM sind einfach zu teuer. Für das Geld könnte man großformatige Farbposter erwarten. Doch wer das Buch kauft, in dem ausreichend schöne Bilder vorhanden sind, muß das Angebot ja nicht wahrnehmen. Ich empfehle die Ausstellungskataloge der Forschungsgruppe "Komplexe Dynamik" an der Universität Bremen zu bestellen. Anschrift und Details auf Seite 315 im Buch.

Ich kann das Buch jedem empfehlen, der mit seinem Computer mehr machen will als nur Ballerspiele oder "Pac Man". Man lernt etwas, das sicher auch in der Schule zu verwenden ist. Gerade die experimentelle Mathematik auf dem Heimcomputer kann vielleicht dazu beitragen, die (völlig unverständliche) Abneigung gegen Mathematik zu überwinden. Schwierig ist die in diesem Buch benutzte Mathematik nicht.

Rainer W. Gerling

Der Sinclair unter Kontrolle

Verlag Gigasoft
192 Seiten DIN A4
69,– DM

Dieses Buch ist eine Neuheit für den QL-User. Schlägt man es auf, so stößt man auf ein sehr gut und übersichtlich gegliedertes Inhaltsverzeichnis. Schon Kapitel 1 zeigt, für wen das Buch gedacht ist: Es ist dies der Kenner der Maschinensprache. Auf einigen Seiten wird kurz und knapp der Prozessor 68000/68008 erläutert. Diese Darstellung ist übersichtlich und informativ, doch eigentlich überflüssig, denn wer Maschinensprache kennt, bedarf einer solchen Einführung nicht.

Mit dem 2. Kapitel beginnt der eigentliche Wert des Buches. Hier wird eine Einführung in das QDOS-Betriebssystem des Sinclair-QL behandelt. Wichtige Punkte sind dabei das Multitasking und die Aufteilung des QL-Speichers.

Kapitel 3 erläutert die sogenannten "Traps" und deren Aufgaben (Traps und die Routinen des Betriebssystems).

In Kapitel 4 folgt dann endlich eine genaue Einzelbeschreibung der Arbeitsweise der Traps. Es werden die Aufrufparameter behandelt. Anschließend folgen Angaben, welche Register angesprochen werden.

Vektorisierte Hilfsprogramme beschreibt der Autor in Kapitel 5. Vektorisiert bedeutet, daß die Adressen der Hilfsprogramme in einer Tabelle stehen. Zur Anwendung muß der Benutzer dann nur noch den Standort in der Tabelle kennen. Dies folgt in Kapitel 6. Auch werden diese Hilfsprogramme hier noch genauer erläutert.

Kapitel 7 verläßt dann die Theorie und stellt 2 Basic-Programme und 2 Maschinenprogramme vor. "Trapper", in Basic geschrieben, ist eine kleine Einführung in die Maschinensprache. Sie ermöglicht die Eingabe von MC-Routinen und zeigt deren Auswirkung auf die Register.

Die restlichen Kapitel befassen sich mit dem maschinenmäßigen Standort von Super Basic und dem Microdrive.

Zusammenfassend läßt sich zu dem auf den ersten Blick etwas teuer erscheinenden Buch sagen, daß es viel Informationen bietet. Außerdem ist zu bedenken, daß dieses Buch nur in einer kleinen Auflage erscheinen kann und die Herstellungskosten erheblich sind.

M. L. Schirmer

Das Buch kann über den CK-Verlag bezogen werden.

Microsoft-Basic auf dem QL

Für den CP/M-Rechner und auch für IBM-PCs oder Kompatible gibt es sehr viele Software. Erstaunlich ist, wieviele Programme davon in Basic geschrieben sind. Insbesondere in den USA gibt es viele Basic-Spiele für den IBM PC. Auch eines der besten Public-Domain-DFÜ-Programme „PC TALK" ist in Basic geschrieben. Ich kenne auch einen guten 8086-Assembler (CHASM), der in Basic geschrieben ist. Eine Anpassung dieser Programme an den eigenen Rechner ist die Arbeit wert, da die Programmquellen meistens kostenlos sind. Auch CP/M-Basic-Programme erleben gerade durch die Schneider Computer und den Commodore 128 einen zweiten Frühling.

Der QL ist für eine solche Anpaßarbeit optimal geeignet. Mit der Möglichkeit, Prozeduren und Funktionen zu definieren, kann man die häufigsten Basic-Befehle aus dem Microsoft-Basic simulieren. Das Listing zeigt die Wichtigsten. Die Grafik- und Sound-Befehle müssen von Hand individuell angepaßt werden. Folgende Vorarbeit sollte jedoch schon gemacht sein:

1. Alle TABs müssen durch TOs ersetzt werden.
2. Alle Hochkommas, die Kommentare abgrenzen, müssen durch REMs ersetzt werden.
3. Falls es fehlt, ist nach allen THEN-Befehlen das GOTO einzufügen.
4. Nach allen NEXT-Befehlen müssen zusätzlich die Namen der Schleifenvariablen angefügt werden.
5. Vor allen ELSE-Befehlen ist ein Doppelpunkt einzufügen.

Drei Konstruktionen kann man nur von Hand ändern. 1. Die Konstruktion

```
10 WHILE bedingung
20
30 wend
```

muß durch

```
10 REPeat name
20 IF bedingung THEN EXIT
   name
30
40 END REPeat name
```

ersetzt werden. 2. Wenn bei einer FOR-NEXT-Schleife nach dem FOR-Statement schon der erste Befehl der Schleife folgt, dann muß der Befehl in eine neue Zeile. Andernfalls gibt es Probleme mit der Kurzform der FOR-NEXT-Schleife (siehe QL-Handbuch). Ein Beispiel:

```
10 FOR i = 1 TO 10: a=b**45

20
30 NEXT i
```

muß durch

```
10 FOR i = 1 TO 10
11 a=b**45
20
30 NEXT i
```

ersetzt werden. 3. Alle Abfragen mit Zeichen wie z.B. "IF a$ >= "a" THEN" müssen durch den ASCII-Code ersetzt werden. Aus obigem Beispiel ergibt sich deshalb "IF CODE (a$) >=97 THEN". Das liegt an der Sortierreihenfolge des QL. Es werden nicht die ASCII-Codes verglichen, sondern die Zeichen in der Reihenfolge „AaBbCc" usw. (siehe Handbuch Begriffe/Stringvergleich Seite 48 bzw. 52).

Im Listing sind die wichtigsten Befehle umgesetzt, die man für einfache Befehle benötigt. Die Funktion INSTR (a$, b$) berücksichtigt keine Groß- und Kleinschreibung. Außerdem überdecken einige Funktionen/Prozeduren nicht die gesamte Breite der Möglichkeiten. Es sind manchmal nur die wichtigsten Parameter angegeben. Das ist aber dann diejenige Form, die am häufigsten benutzt wird.

Es ist sicherlich möglich, noch Verbesserungen vorzunehmen. Vielleicht fehlt auch der eine oder andere Befehl. Das vorgestellte Listing soll nur den Weg zeigen und ein Anfang sein.

Rainer W. Gerling

```
1 REMark Anpassung QL SuperBASIC an
2 REMark Microsoft Basic
3 REMark (c) 1986 by Rainer W. Gerling
4 REMark Version 1.1
5 REMark ******************************
6 DEFine FuNction LEFT$(a$,b)
7 IF b=0 THEN RETurn ""
8 IF b>LEN(a$) THEN RETurn a$
9 RETurn a$(1 TO b)
10 END DEFine
11 REMark ******************************
12 DEFine FuNction RIGHT$(a$,b)
13 IF b=0 THEN RETurn ""
14 IF b>LEN(a$) THEN RETurn a$
15 RETurn a$(LEN(a$)-b+1 TO)
16 END DEFine
17 REMark ******************************
18 DEFine FuNction INSTR(a$,b$)
19 RETurn b$ INSTR a$
20 END DEFine
21 REMark ******************************
22 DEFine FuNction MID$(a$,A,b)
23 IF b=0 THEN RETurn ""
24 IF A>LEN(a$) THEN RETurn ""
25 RETurn a$(A TO A+b-1)
26 END DEFine
27 REMark ******************************
28 DEFine PROCedure COLOR (A,b)
29 INK A : PAPER b
30 END DEFine
31 REMark ******************************
32 DEFine PROCedure PLAY(a$)
33 REMark Ist nur eine primitive Losung
34 BEEP 20000,10,100,1000,10,5
35 END DEFine
36 REMark ******************************
37 DEFine FuNction VAL(X$)
38 RETurn X$
39 END DEFine
40 REMark ******************************
41 DEFine FuNction STR$(X)
42 RETurn X
43 END DEFine
44 REMark ******************************
45 DEFine FuNction SGN(X)
46 IF X>0 THEN RETurn 1 : ELSE IF X=0 THE
 N RETurn 0 : ELSE RETurn -1
47 END DEFine
48 REMark ******************************
49 DEFine FuNction asc(X)
50 RETurn CODE(X)
51 END DEFine
52 REMark ******************************
53 DEFine PROCedure files(a$)
54 DIR a$
55 END DEFine
56 REMark ******************************
57 DEFine FuNction space$(X)
58 RETurn FILL$(" ",X)
59 END DEFine
60 REMark ******************************
61 DEFine FuNction string$(X,a$)
62 RETurn FILL$(a$(1),X)
63 END DEFine
64 REMark
65 REMark ******************************
```

Grafische Benutzeroberflächen für den QL

Das erste Desktop à la GEM für den QL brachte das britische Softwarehaus Eidersoft heraus. Mittlerweile gibt es zwei ähnliche Produkte aus Deutschland: "E A S E" (Easily Applicable System Environment) von Gigasoft und "J A M" (Job Application Monitor) von zwei fränkischen Autoren. Wir haben beide Programme getestet.

Das Programm "E A S E" benötigt ca. 64 KByte, womit beim Grund-QL schon der meiste Speicherplatz belegt ist. Deshalb wird es beim Verlassen sofort gelöscht und nach einem erneuten Aufruf wieder geladen. In einem QL mit Speichererweiterung ist es ständig parat. "E A S E" wird über Pull-Down-Menüs gesteuert: auf dem Bildschirm befindet sich ein Pfeil, der mit Maus, Joystick oder Cursor-Tasten in ein Menüfeld gebracht wird. Drückt man nun auf Maustaste, Feuerknopf oder Leertaste, wird der angesteuerte Menüpunkt ausgeführt. Damit der Bildschirm frei bleibt, gibt es am oberen Rand nur wenige Menüpunkte, von denen bei Anwahl ein Untermenü gleich einem Rollo herunterfällt. "E A S E" bietet vier Begriffe: DESK, FILE, OPTIONS und EXIT. Der letzte Punkt dient zum Verlassen des Programms.

Hinter DESK verbergen sich vier Unterpunkte. Der erste, ABOUT E A S E, ist der gleiche Betrug am Benutzer wie beim GEM auf dem Atari ST. Hinter CALCULATOR verbirgt sich ein Taschenrechner mit UPN-Eingabe wie bei HP-Taschenrechnern. Er verfügt über etliche wissenschaftliche Funktionen sowie eine Rechengenauigkeit, die Super-Basic entspricht. Was ich allerdings vermißt habe, sind Hexadezimalzahlen, die ein Computerbenutzer doch häufig braucht. Die Bedienung des Taschenrechners mit der Maus ist nur etwas für verspielte Zeitgenossen, eine Eingabe über Tastatur wäre angebracht. Dann gibt es noch den Punkt GAME, ein einfaches Spiel, um die Zeit totzuschlagen. Gut ist, daß Taschenrechner und Spiel mehrfach (und gleichzeitig) angewählt werden können. Der vierte Punkt PANEL erlaubt das Einstellen diverser Systemparameter (z.B. Default-Laufwerk).

Der Menüpunkt FILE macht so ein Desktop-Programm erst richtig interessant, da beim QL das Kopieren von Files ziemlich umständlich ist. Die Befehle sind lang, und auch das Löschen bedeutet viel Tipparbeit. Hier werden nun alle zu kopierenden/löschenden Files markiert (d.h. mit Pfeil anfahren und anklicken), anschließend wählt man ganz einfach COPY oder DELETE. Ein "Klick" und der Rest geht von alleine. Was leider fehlt, ist der Unterpunkt RENAME. Es gibt aber die Punkte FORMAT (den Namen der Cartridge muß man wirklich eintippen) und START. Damit kann man einen Job starten oder ein anderes File listen bzw. drucken. Auch ein Super-Basic-Programm kann gestartet werden. So wird die Fileverwaltung zu einem Kinderspiel, die lästige Tipperei entfällt. Cartridges oder Floppys, die man immer schon mal aufräumen wollte, werden jetzt von allem Datenschrott befreit.

Der letzte Punkt ist OPTIONS. Hier kann man einstellen, ob man seine Files als Icons oder als Text sehen möchte. Icons sind kleine Bilder, die den Filetyp angeben. QUILL-Files sind typischerweise kleine beschriebene Blätter, EASEL-Files kleine Diagramme usw. In der Option "Text" werden neben Filename auch Länge und Erstelldatum angegeben. Im normalen QDOS gibt es kein Datum, während z.B. Toolkit II alle Files damit versieht. Die Files können in diesem Modus auch wahlweise nach Namen, Länge und Datum sortiert werden.

Kommen wir zum "J A M" aus Franken. Hier sind alle Fenster fest vorgegeben. Ein Verschieben ist auch nicht erforderlich, da alle Menüpunkte nur einmal angewählt werden können. Was ist also der Unterschied zu "E A S E"? Zunächst gibt es kein Spiel. Aber wer ein Spiel will, der kauft kein Desktop. "J A M" ist mit 27 KByte lang, also wesentlich kürzer. Der Taschenrechner bietet weniger Funktionen, ist aber über Tasten zu bedienen. Es gibt einen Kalender, der den jeweils aktuellen Monat anzeigt. Man kann ihn vor- und zurückblättern, was sich bei der Terminplanung als praktisch erweist.

"J A M" verfügt am linken Rand über fünf Symbole für die unterstützten Laufwerke: zwei Floppys, zwei Microdrives und eine RAM-Floppy. Diese RAM-Floppy ist als schneller Zwischenspeicher eine große Hilfe beim Kopieren von Files. Hat man nur ein Laufwerk, so kopiert man von der ersten Diskette auf die RAM-Floppy und von da auf die zweite Diskette. Ein Zwischenspeichern auf Microdrive entfällt, der Kopiervorgang wird viel schneller.

Das File-Menü bietet auf den ersten Blick sehr viel mehr, als es später hält. Hat man sich näher damit beschäftigt, erkennt man, daß nur die Organisation anders ist. Alles, was mit "E A S E" möglich ist, geht auch mit "J A M". Ein wichtiger Pluspunkt ist aber die Option RENAME zum einfachen Ändern von Filenamen.

Während "E A S E" ausschließlich Fileverwaltung und Desktop-Funktionen bietet, gibt es bei "J A M" eine gute Verwaltung von Jobs. Im JOB-Menü können sie alle gelistet, suspendiert oder gelöscht werden, auch Prioritäten lassen sich ändern ("J A M" nie auf Priorität Null setzen). Da QDOS keine Befehle zur Job-Verwaltung bietet, ist dies ein dickes Plus.

Unter der Rubrik "Options" bietet "J A M" einen Taschenrechner, einen Kalender, Memory Info und einen Notizzettel. Eine kurze Notiz wird beim nächsten Start automatisch angezeigt. Der Taschenrechner bietet zwar nur die Grundrechenarten und Wurzelziehen, ist dafür aber über die Tastatur zu bedienen. Auch hier fehlen die so wichtigen Hexadezimalzahlen. Der Kalender zeigt beim Einsprung immer den aktuellen Monat, man kann nun beliebig vor- und zurückblättern. Es gibt auch die Möglichkeit, die Geschwindigkeit des Pfeils einzustellen. Benutzt man einen Joystick, so dauert es ein Weilchen, bis man die optimale Geschwindigkeit gefunden hat.

"J A M" ist kopiergeschützt, d.h. man kann zwei Kopien ziehen, dann weigert sich das Kopierprogramm, weitere zu produzieren. Alle anderen Methoden (QLone, COPY, "J A M" selbst) haben keine lauffähige Version erzeugt.

Beim Vergleich der beiden Desktop-Programme schneidet "J A M" besser ab. Es ist praxisorientierter und nicht so verspielt. Auch die Tatsache, daß es nur etwa die Hälfte an Speicherplatz benötigt, ist nicht unwichtig.

Vergleicht man die Preise ("J A M" ca. 69,– DM, "E A S E" ca. 222,– DM), so ist zu berücksichtigen, daß "E A S E" mit anschlußfertiger Maus und Giga-Basic (eine Basic-Erweiterung, die alleine ca. 50,– DM kostet) geliefert wird. Die Gigasoft-Maus belegt allerdings neben dem Joystick-Port CTRL1 auch den ROM-Port, was für diverse Compiler- und Toolkit-Besitzer einige Probleme mit sich bringen dürfte.

Weitere Informationen:

"E A S E": ABC Electronic
Andreas Budde
Hügelstraße 10-12
4800 Bielefeld 1

"J A M": T.S. Datensysteme
Dennerstraße 45
8500 Nürnberg 30
Computerstudio
Kreuzstraße 13
8000 München 2

Tricks und Tips für den QL

FOR und REMark

Das Super-Basic des QL kennt zwei Varianten der FOR-END FOR-Schleife: die kurze und die lange Version. Beim Umschreiben von Basic-Programmen von anderen Rechnern kann es mit der kurzen Version Probleme geben. Diese Version liegt immer dann vor, wenn nach dem FOR in der Zeile noch ein weiteres Statement folgt und nicht etwa nur dann, wenn kein NEXT oder END FOR folgt. Eine Fehlerquelle ergibt sich, wenn auch ein REMark ein Statement ist. Um dieses zu demonstrieren, zeigt Listing 1 ein kleines Beispiel, das diesen Punkt verdeutlicht. Obwohl ein REMark den Programmablauf nicht beeinflussen sollte, tut es dies jedoch ganz entscheidend. Viele Basic-Programmierer schreiben einige Statements auf anderen Rechnern bereits hinter das FOR. Hier muß man beim Umschreiben auf den QL sehr aufpassen, denn solche Fehler sind manchmal schwer zu finden.

Löschen von Prozeduren und Funktionen

Hat man Prozeduren oder Funktionen im Programm, so kann es recht häufig vorkommen, daß man die eine oder andere auch mal löscht. Dazu verwendet der normale Programmierer den Befehl DLINE. Aber Vorsicht: Durch das Löschen der Programmzeilen wird der Prozedureintrag in den Systemvariablen nicht gelöscht. Das heißt, der Rechner erkennt den Namen der Prozedur/Funktion immer noch, findet sie aber natürlich nicht mehr.

Geben Sie dazu eine kleine Prozedur ein und löschen diese Zeilen mit DLINE. Verwenden Sie anschließend den Prozedurnamen als Variable. Jetzt werden verschiedene Fehlermeldungen erscheinen. In Einzelfällen kann der Rechner sogar abstürzen. Die einzige Hilfe: Das Programm saven, dann einen Reset durchführen und das Programm neu laden.

Prozedur- und Funktionsnamen sollte man bei der ersten Eingabe immer mit Großbuchstaben schreiben. Sie werden dann bei weiteren Eingaben unabhängig von der Schreibweise groß geschrieben. Ist der Name einmal mit Kleinbuchstaben eingegeben, wird er immer mit Kleinbuchstaben geschrieben. Gibt man den Namen gemischt ein, erfolgt die Schreibweise auch hier entsprechend. Eine Eingabekürzung (wie z.B. bei REMark) läßt sich so aber nicht garantieren.

Im Block-Befehl gibt es einige kleine Fehler. Die Breite muß kleiner als 512 sein. Gibt man 512 an, so passiert nichts, denn 512 wird als Null angegeben. Die maximale Breite eines Blocks ist 511 Pixel. 513 wäre Breite 1. Für die Höhe ist 256 der Maximalwert. Bei 257 gibt es eine klare Fehlermeldung.

Der Point-Befehl

Im deutschen QL (ROM-Version MGG) gibt es einen Fehler: Der Befehl POINT zeichnet nicht ein Pixel, sondern zwei. Das zweite ist oberhalb um ein Pixel nach links versetzt. Da alle anderen Zeichen-Routinen (LINE, CIRCLE, BLOCK usw.), falls sie nur ein Pixel zeichnen müssen, auf die POINT-Routine zurückgreifen, haben diese ebenfalls das Problem. LINE 50,50 TO 50,50 arbeitet fehlerhaft, LINE 50,50 TO 50,51 funktioniert jedoch richtig. Da das Pascal von Computer One ROM-Routinen anspringt, gibt es den Fehler auch hier.

SELECT in Prozeduren

Es wurde berichtet, daß bei einigen englischen ROM-Versionen es nicht möglich ist, eine Übergabevariable in einer Prozedur als SELECT-Variable zu benutzen. Tests auf dem deutschen QL haben gezeigt, daß es hier geht. Benutzer englischer

Listing 1 demonstriert den Unterschied zwischen der kurzen und langen FOR-Schleife. Der dritte Teil zeigt, wie ein REMark die Schleife beeinflußt.

```
100 CLS
110 REMark Demo lange Version
120
130 FOR i= 1 TO 10
140 PRINT i,i*i,i^3,i^4
150 END FOR i
160 PAUSE:CLS
170
180 REMark Demo kurze Version
190
200 FOR i=1 TO 10: PRINT i,i*i,i^3,i^4
210 PAUSE:CLS
220
230 REMark Demo lange Version mit REMark
240
250 FOR i= 1 TO 10: REMark
260 PRINT i,i*i,i^3,i^4
270 END FOR i
```

Listing 2 zeigt, wie man eine LLIST-Prozedur mit Druckersteuerung schreiben kann. Um Platz zu sparen, sollte man die REMarks am Ende einer jeden Zeile nicht mit abtippen. Zeile 240 kann ganz entfallen. Die Einstellung des rechten Randes läßt sich gegebenenfalls auch einbauen.

```
100 REMark LLIST für Sinclair QL
110 REMark Druckersteuerzeichen fur TA Gabriele 9009
120 REMark R.W. Gerling
130 REMark Juni 1986
140 DEFine PROCedure LLIST
150 LOCal esc$:esc$=CHR$(27)              REMark Codes fur ESC und CR
160 OPEN#99,ser1                          REMark Kanal 99 auf Schnittstelle 1
170 TRA 1                                 REMark Ubersetzen auf deutschen ASCII
180 PRINT#99,esc$;CHR$(13),"P",           REMark Reset des Druckers
190 PRINT#99,esc$,"a",                    REMark Auto Carriage Return ein
200 PRINT#99,esc$;CHR$(5),                REMark linken Rand setzen
210 rand=4:PRINT#99,CHR$(rand),           REMark auf 4te Schreibstelle
220 PRINT#99,esc$;"9"
230 PRINT#99,esc$;CHR$(22);CHR$(2),       REMark deutschen Zeichensatz fur
240                                       REMark Schreibmaschine
250
260 LLIST#99,1000 TO
270 END DEFine LLIST
```

Uta Jäkel & A. Klintworth GbR

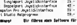
Hard- & Softwarevertrieb
Harschhorst 2

2732 Klein - Meckelsen
Telefon: 04282 / [illegible]

Wir wünschen allen Kunden, Interessenten und Freunden

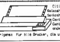
ZX Spectrum 48 KB	249.-
ZX Spectrum plus 48 KB	329.-
ZX Spectrum 128 KB	449.-
Saga 1 Tastatur	129.-
Saga 3 Tastatur	249.-
Dk tronics Tastatur	139.-
Kempston E Centronicsinterface	155.-
ZX Lprint III Druckerinterface	175.-
Multiface One (neue Ausführung)	155.-
Dk Singleport Joystickinterface	29.-
Dk Doppelport Joystickinterface	45.-
Kempston Pro Joystickinterface	59.-
Art Studio Malprogramm	59.-
Discovery 180 KB	399.-
Discovery 360 KB	509.-
Discovery 720 KB	799.-
Discovery 1440 KB	1199.-
Einbaulaufwerk 180 KB	199.-
Einbaulaufwerk 720 KB	399.-
Beta Disk Controller 5 01	328.-
Beta Disk Komplettsystem 1 MB ab	749.-
SpecDrum Schlagzeug-Synthesizer	149.-
Sprachsynthesizer	89.-
Dk 3-Kanal Soundsynthesizer	189.-
Maus incl. Interface ab	269.-
Tasword III (Cartridge oder Disk)	89.-

Achtung! Wir führen auch Software für den ZX Spectrum 128 KB !!!



QL 128 KB englische Ausführung	429.-
QL 128 KB deutsche Ausführung	459.-
Diskcontroller (Software im Eprom) ab	299.-
Disk Komplettsystem (1 MByte) ab	653.-
Maus incl. Interface und Software	220.-
Super Diskcontroller + Centronicsinterface + TOOLKIT II + 512 KB auf einer Karte	793.-
512 KB Speichererweiterung mit durchgeführtem Bus (einfach ansteckbar)	389.-
QL 128 KB (englische oder deutsche Ausführung) + 512 KB Speichererweiterung	799.-
Softwarepaket für Betriebe (Fibu, Adressverwaltung, Kalkulator, Serienbriefe, Lohn- und Gehaltsabrechnung, Lagerverwaltung, Faktura, läuft nur mit Disk)	890.-
Monitors (grün, Bernstein) ab	248.-
CUB 653 Super Farbmonitor	1090.-
QL Centronicsinterface ab	139.-
QSound / QPrint Interface	138.-
QL ART Super Grafikprogramm (Font Editor, Zoom, 2 Screens à 512 x 256 Pixel, sehr benutzerfreundlich, selbstvorführende Demo lieferbar)	98.-
QL ART+ wie **QL ART** jedoch wesentlich umfangreicher (läuft nur mit Zusatzspeicher, Auflösung und Grafikseite 720 x 564 Pixel, Textured Fill, auf Epson komp. Druckern als **DIN A4** oder **DIN A2** - Poster ausdruckbar)	148.-

Cartridges	Stk.: 7,50	12 Stk.: 85.-	20 Stk. in Cartridge - Box 158.-
Cartridge Box	20.-	3,5" - 50er Disketten Box mit Schloß	28.-
3,5" - Disk No Name 135 TPI	Stk.: 5,50	10 Stk.: 5[illegible].-	50 Stk.: 225.-
3,5" - Disk Scotch SSDD 135 TPI	Stk.: 7,50	10 Stk.: 70.-	50 Stk.: 329.-
3,5" - Disk Scotch DSDD 135 TPI	Stk.: 9.-	10 Stk.: 75.-	50 Stk.: 350.-



Citizen 120 D (IBM + Epson kompatibel)	729.-
Gabriele HS15 Typenraddrucker (RS232 + Centronics)	659.-
Seikosha SP 1000 (A / AS / I / VC)	749.-
Centronics GLP II (RS232 + Centronics, 100 Z/s)	598.-
Centronics GLP II + Aufsatztraktor	648.-
andere Drucker (Star, Oki, usw.)	auf Anfrage

Übrigens: Für alle Drucker, die wir verkaufen, können wir auch Farbbänder liefern !

des Hauses ein frohes und gesegnetes Weihnachtsfest !!!

Lieber Kunde, die Preise können sich zwischen Layouterstellung und Veröffentlichung dieser Anzeige schon geändert haben. Daher:

Ruf doch mal an!

Versand nur per Nachnahme oder Vorauskasse ! Selbstabholung und
Überführung von Artikeln nur nach vorheriger telefonischer Terminabsprache !
Alle Preise zzgl. Versandspesen zum Selbstkostenpreis !
Unsere Gesamtpreisliste erhalten Sie gegen 2,- DM in Briefmarken !

Übrigens! Wir exportieren auch ins Ausland !!!

Diese Anzeige wurde komplett und in Originalgröße mit unserem Graphikprogramm ART+ erstellt !

QLs sollten einmal berichten ob es bei Ihnen geht

Das Listen von Super Basic Programmen artet falls man den Drucker auch an anderen Computern benutzt häufig in eine ziemliche Spielerei mit den DIP-Schaltern aus. Eine einfache Lösung bietet dazu die Prozedur LLIST. Sie stellt den Drucker softwaremäßig auf die entsprechenden Parameter ein. Am Drucker sollte Auto Carriage Return (CHR$(13)) eingegeben werden, da der QL nur ein Linefeed (CHR$(10)) sendet. Auch die Einstellung der Breite muß am Drucker erfolgen, da der Befehl WIDTH bei Listings nicht funktioniert. Alte Spectrum Fans können am Drucker auch 32 Zeichen pro Zeile eingeben. Zusätzlich gibt es noch viele weitere Möglichkeiten etwas einzustellen. Listing 2 zeigt eine mögliche Form. Damit man nicht ständig den LLIST Teil mit auflistet sollten die Programme immer an Zeile 1000 beginnen. LLIST listet dann nur ab Zeile 1000.

Hilfsroutinen

Man kann natürlich unterhalb Zeile 1000 auch noch andere wichtige Prozeduren und Funktionen ablegen. Listing 3 zeigt drei Beispiele. Es sind dies der Befehl CAT für Katalog sowie die Prozeduren ZEIT und TAG. Der Befehl CAT verlangt eine Laufwerksnummer (1 für MDV1_, 2 für MDV2_, usw.) und gibt dann den Inhalt der Cartridge im Laufwerk aus. Mit ZEIT wird die Uhrzeit und mit TAG das Datum gestellt. ZEIT und TAG entsprechen in der Funktion den Kommandos TIME und DATE unter MS-DOS oder CP/M. Es gibt beliebig viele Möglichkeiten, hier der Phantasie freien Lauf zu lassen und seine eigene Hilfsroutinen zu schreiben. Und alles geht in Basic vonstatten.

Rainer W. Gerling

Listing 3 zeigt drei Prozeduren: CAT, TAG und ZEIT. CAT erlaubt einen einfachen Katalog einer Cartridge und mit TAG und ZEIT läßt sich sehr einfach Datum und Uhrzeit setzen.

```
100 REMark  Hilfsroutinen
110 REMark  Base und Anfagune
120 REMark  R W Gerling
130 REMark  Juni 1986
140 REMark ****************
150 DEFine PROCedure CAT (drive)
160 PRINT "Katalog von Laufwerk ";drive;"
170 DIR "mdv"&drive&"_"
180 END DEFine
190 REMark ***********
200 DEFine PROCedure ZEIT
210 LOCal a$,n$,a,b,c,d,e
220 a$=DATE$: a=a$(1 TO 4): b=(a$(6 TO 8) INSTR
    "JanFebMarAprMayJunJulAugSepOctNovDec"+2)/3
    c=a$(10 TO 11): a$=a$(13 TO 17)
230 PRINT "Alte Zeit ";a$
240 INPUT "Neue Zeit ";n$: IF n$="" THEN PRINT a$: RETurn
250 d=n$(1 TO 2): IF d<0 OR d>23 THEN GO TO 240
260 e=n$(4 TO 5): IF e<0 OR e>59 THEN GO TO 240
270 SDATE a,b,c,d,e,0
280 PRINT
290 END DEFine
300 REMark ***********
310 DEFine PROCedure TAG
320 LOCal a$,n$,a,b,c,d,e,f
330 a$=DATE$: a=a$(1 TO 4): b=(a$(6 TO 8) INSTR
    "JanFebMarAprMayJunJulAugSepOctNovDec"+2)/3
    c=a$(10 TO 11)
340 PRINT "Altes Datum ";c;".";b;".";a
350 INPUT"Neues Datum ";n$: IF n$="" THEN PRINT c;
    ".";b;".";a: RETurn
360 c=n$(1 TO 2): IF c<1 OR c>31 THEN GO TO 350
370 i=c<10
380 b=n$(4-i TO 5-i): IF b<1 OR b>12 THEN GO TO 350
390 a=[illegible](b<10)
400 a=n$([illegible] TO 8-i): IF a<[illegible] OR a>[illegible] THEN GO TO 350
410 [illegible] DATE$: d=a$(13 TO 14): e=a$(16 TO 17)
    f=a$(19 TO 20)
420 SDATE a,b,c,d,e,f
430 PRINT
440 END DEFine
450 REMark ***************
460 CLS
470 PRINT "QDOS VERSION ";VER$
480 TAG: ZEIT
```

Spectrum Shadow ROM Disassembly

Von Gianluca Carri
Verlag Melbourne House
ISBN 0-86161-191-8

Vor einiger Zeit ist im englischen Verlag Melbourne House ein Disassembly des Interface 1 ROMs erschienen. Für jeden, der in Maschinensprache programmiert und der die M-Code Routinen im Interface 1 nutzen will, ist dieses Buch fast schon ein Muß.

Im ersten Teil wird ausführlich dargestellt, wie man neue Befehle für das Sinclair Basic schreiben kann. Danach wird ausführlich eine 1903 Byte Basic-Erweiterung besprochen. Die neuen Befehle sind so gut, daß sie alleine den Kaufpreis schon wert sind. Im einzelnen handelt es sich dabei um einen Doppel POKE Befehl, einen String POKE Befehl sowie einen erweiterten BEEP-Befehl. Doch das ist nur der Anfang. Die Knüller sind ein pseudo Random File Befehl, der es erlaubt, gezielt einen Sektor vom Microdrive zu laden. Die Syntax ist READ #s,n, wobei s der Strom und n die Nummer des Sektors im File ist. Das File muß vorher mit OPEN # geöffnet werden. Dazu gibt es noch einen Befehl, mit dem man ein bestehendes File auf der Cartridge erweitern kann. Es heißt RESTORE #s (s ist wieder die Strom-Nummer) und ermöglicht es, am Ende des Files weiterzuschreiben. Sehr hilfreich ist auch der erweiterte CAT Befehl (er gibt Filetyp, Startadresse, Länge und Autostartzeile an), der Memory Dump-Befehl und die Erweiterungen des RS 232 Kanals. Mit am besten finde ich den neuen EDIT Befehl. Damit kann man ohne vorhergehendes LIST eine Zeile editieren. Man muß nur die Zeilennummer angeben.

Es versteht sich von selbst, daß die Befehlserweiterung für beide Versionen des Interface 1 ROMs angegeben ist. Und sollte es einmal eine dritte Version geben, so gibt es Hinweise, wie man die Maschinensprache den Routinen anpaßt. Alle Routinen sind auch komplett im Quellcode gelistet und ausführlich kommentiert.

Der zweite Teil des Buches, das einmal nicht von einem Engländer geschrieben wurde, ist ein komplettes kommentiertes Listing des Shadow ROM im Interface 1. Gelistet ist hier das ROM der Version 1. Die Änderungen in der Version 2 werden im Anhang erläutert. Die Qualität des Disassemblys steht der in "The Complete Spectrum ROM Disassembly" von Logan/O'Hara im gleichen Verlag erschienen nicht nach. Die Kommentare und Erklärungen sind ausführlich und gut.

Wer nicht weiß, welches ROM sich in seinem Interface 1 befindet, der schaue sich die Seriennummern an: bis 87315 ist die Version 1 eingebaut. Ein weiterer kleiner Test ist im Buch beschrieben. Das Buch kostet 8.95 Pfund, wenn man es in England beim Verlag bestellt. Dazu kommen noch 0.80 Pfund Versandkosten. Ich kann das Buch jedem, der in die Details des Interface 1 einsteigen will, wärmstens empfehlen.

An dieser Stelle noch ein Tip zum Bestellen in England: Viele Firmen nehmen keine Euroschecks und wenn sie welche nehmen, dann kommen noch viele Gebühren dazu. Ich benutze zum Bezahlen in England immer eine Kreditkarte. Entweder braucht man eine Visa Karte oder eine Eurocard. Vielleicht hat ja ein Bekannter oder Verwandter eine solche Karte. Das spart Gebühren und macht das Bestellen einfacher. Es geht damit sogar per Telefon und die Rechnung gibt es in DM. Die Anschaffung einer Kreditkarte wegen des Computerhobbys lohnt sich allerdings nicht.

Rainer Gerling

**Ihr direkter Draht
zur Redaktion:
Telefon 0 72 52 / 4 29 48
Für Briefe:
Computer Kontakt
Postfach 16 40
7518 Bretten**

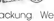
Wir machen ein auf !

Fehlt Ihnen noch eine Ausgabe von CK

vom **85**er Jahrgang ? Oder mehrere ?

Oder sogar alle ?

Das muß nicht sein ! Wir schnuren jetzt ein . Und da geht echt die ab. Da ist das drin und Ja sogar Und mit setzen wir dem ganzen die auf. Das sind **9** Hefte die am Kiosk 40 Mark und 50 Pfennige kosteten.

Und bei uns ? -  Inclusive und Verpackung. Wer jetzt nicht den Coupon » « ist selber schuld.

COMMODORE - NEWS

Hallo Commodore-Freunde!

Wie im Vorwort des Heftes schon angekündigt, wird der Commodore Teil in der CK nicht weiter erscheinen. Nur den letzten Teil des Grafikkurses werden wir in Heft 2-3/87 noch bringen können.

Seit der ersten Ausgabe der CK im Mai 1984 war der Commodore Teil dabei. Handelte es sich damals noch um unsere Hauptlesergruppe, so hat sich dies immer mehr verlagert. Die übergroße Kokurrenz ist zu stark, das hat sich gerade in den letzten beiden Ausgaben gezeigt, als die Commodore Leser immer weniger wurden. Diese Situation zwingt uns dazu, den Commodore Teil aufzugeben. Wir finden das zwar schade, aber ein kleiner Verlag wie wir muß sich absolut an den Marktgegebenheiten orientieren.

Trotzdem auch in diesem Heft haben wir noch einige interessante Beiträge auf Lager. Wie immer gibt es Programme, neue Software, Buchrezensionen sowie Tips und Tricks. Auch für dieses letzte Heft haben wir uns für unsere Leser noch einmal ins Zeug gelegt.

An dieser Stelle möchte ich mich bei all den Lesern bedanken, die trotz der großen Commodore Zeitschriften Konkurrenz uns treu geblieben sind. Da ist mancher dabei, der die CK seit ihren Anfängen kennt und nun – vielleicht etwas wehmütig – auf sie verzichten muß. All diesen Lesern wünschen wir alles Gute und natürlich weiterhin viel Spaß mit ihrem Computer, schließlich ist besagte Konkurrenz auch nicht gerade schlecht.

Thomas Cherek

C 64 und PC 128

S/5 Modul 30,00 DM
Supersoftwarepaket I 16,00 DM
10 Disketten SS/DD 14,00 DM
Superpreis
S/4 Modul 23,00 DM
Modul 2000 45,00 DM
Modul 3000 68,00 DM
Modus S-100 69,00 DM
Super Hardcopy Modul 46,00 DM
S/6 Modul 36,00 DM
S/1 Modul 18,00 DM
Turbo Tape
S/2 Modul 18,00 DM
Profi Betriebssystem 49,00 DM
Betriebssystem umschaltplatine 128 38,00 DM
Leerplatine 9,90 DM

Klemmer & Schulte Electronic
5030 Hürth Efferen

Boulder Dash III

Handlungsweise und Story des Spiels sind recht schnell erklärt, doch vielleicht gerade wegen seiner simplen Handlung fesselt einen „Steinschlag" (dritter Teil), bis man nicht mal mehr Kaffee kochen oder eine Zigarette rauchen will.

Getestet wurde die Cassettenversion für den Commodore 64. Man kann jedoch davon ausgehen, daß es Versionen für so ziemlich alle gängigen Computertypen gibt.

Es geht im Spiel darum, daß man mit einem kleinen Männchen namens Rockfort durch einen Screen flitzen und alle dort herumliegenden Diamanten einsammeln muß, um ins nächste, in der Regel wesentlich schwerere Bild zu gelangen. Wer jetzt weiterblättert und "Viel zu einfach" sagt, ist selbst schuld und hat sich schwer getäuscht. Bei seiner Suche nach den Diamanten muß Rockfort nämlich auf eine Menge achten. Wer beim Graben nicht aufpaßt oder gar zu viele Steine lockert, wird recht schnell erschlagen bzw. blitzschnell eingesperrt. Doch damit nicht genug. Als weitere Schikane sausen noch einige Gestalten (Müller und Augen) durch die Höhle, von der man übrigens immer nur einen kleinen Ausschnitt sieht. Bei Berührung dieser Figuren verliert man eines von drei Leben. Bisweilen beginnt einem Raum nicht genug Diamanten um in die näch ste Höhle zu gelangen. Das muß man zu einem kleine Trick greifen. Verrückt mo ein Auge so erhält man n Ihr 9 Diamanten. Sollten die immer noch nicht ausreiche so findet man in solchen Ra men noch laut blubbernde L va, die man durch Einsperr mit Steinen in Diamanten ve wandeln kann. Es besteht da nur noch das Problem an s ranzukommen. Wem das im mer noch zu einfach erschei der kann sich nach überstand nen 4 Höhlen mit einem Puzz auseinandersetzen, bei dem ein Leben zu gewinnen gib Dies geschieht übrigens alle 5 Punkte; hier erhält man eb falls einen Zusatz-Rockfort.

Die Anzeige aller wichtigen I formationen ist recht gut gest tet. In der obersten Zeile ste während des ganzen Spiels w viele Diamanten man samme muß, wie viele Punkte es p Diamant gibt, der Score un Zusätzlich werden einige sp zielle Informationen angezei So flammt z.B. der Bildschi kurz auf, wenn der Ausgang len ist. Auch mit Gags wurde wie schon bei den Vorgängern nicht gespart. Ist man unsere kleinen Held mit den Knöpf gen zu langsam, so stampft ungeduldig mit dem Fuß.

Sollte man die Vorgäng schon besitzen, so lohnt es si doch, den dritten Teil zu ka fen. Das neue Spiel ist schneller und schwieriger als die beiden ersten Teile. Grafik und Sound sind von der Qualität her unverändert gut, das Menü ist auch weiterhin optimal. Fazit: uneingeschränkt empfehlenswert.

Dirk Zeemis

Parallax

Das Superraumschiff eines außerirdischen Volkes ist auf dem Weg zur Erde. Die Besucher kommen in friedlicher Absicht und nehmen sogar eine Delegation der Erde an Bord auf, bestehend aus den fünf erfahrensten Astronauten der Welt. Die Besatzung des Raumschiffs zeigt sich sehr gastfreundlich. Doch zufällig stolpert einer der Astronauten über gewisse Informationen von einem Computerhirn, die einen bevorstehenden Angriff auf die Erde aufdecken.

Das Superraumschiff mit der gastfreundlichen Besatzung verwandelt sich für die fünf Astronauten zum Deathstar, der die Welt zerstören will. Und wer kann das ändern? Sie ja, ganz genau Sie! Sie schlüpfen nämlich in die Rolle des todesmutigen Raumschiffpiloten, der die vier Astronauten-Kumpels aus den fünf abgegrenzten Zonen des Raumschiffs bergen soll. Um das Tor zum jeweils nächsten Abschnitt zu öffnen, braucht man ein Paßwort, das man den Datenkarten der zu kidnappenden Wissenschaftler entnehmen muß. Am Ende tauscht man dann die vier gekidnappten Wissenschaftler aus den fünf Raumschiff-Zonen gegen die vier Astronauten aus und verschwindet dann in Windeseile in Richtung Heimatplanet Erde.

"Parallax" ist eine ungewöhnliche Mischung aus Action-Adventure und Ballerspiel, gekonnt untermalt mit 3-D-Effekten und recht bombastischer Musik.

Auf der Suche nach den Hangars fliegen Sie mit Ihrem Raumkreuzer Ibis über das gigantisch und gewaltig anmutende Schiff der Fremden. Die Hangars und Hallen, in denen meist ein oder mehrere Computer stehen, die von Robotern oder Wissenschaftlern bewacht werden. Kidnappt man einen Wissenschaftler, so kann man dessen Datenkarte nehmen und diese in das Hauptkontrollsystem der Feinde (den Central Intelligence Unit C.I.U.-Computer oder auch Big One) eingeben, von wo aus man dann einen Buchstaben des Lösungswortes erhält.

Die Computer-Banken oder die Computer-Shops sind auch in vielen Hangars anzutreffen. Durch diese Systeme kann man Geld abheben bzw. ausgeben, wenn man zur Datenkarte ein Paßwort hat. Hat man aber kein Paßwort und möchte Geld bei der Computer-Bank abheben, so behält dieser originelle Computer einfach die Datenkarte.

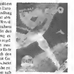
Auf dem Weg von Hangar zu Hangar wird man natürlich von feindlichen Fliegern, Verteidigungsgeschützen, Boden- und Untergrundtruppen belästigt. Dieses Problem läßt sich aber durch Zielen und lockeres Antippen des Feuerknopfes bequem lösen. "Parallax" ist programmiertechnisch ein Referenzwerk für alle Weltraumspiele. Die futuristische Musik wird von der dreidimensionalen Grafik sogar noch übertroffen. Es ist aber auch ein Spiel, dem man nicht gleich ansieht, daß sich eine höhere Handlung dahinter verbirgt. Den Schwierigkeitsgrad konnte man am treffendsten mit "anspruchsvoll" umschreiben.

System	C 64
Hersteller	Ocean

Stefan Bofrach

Nuclear Embargo

"Nuclear Embargo" heißt ein neues Weltraumspiel von Eurogold. Spielidee und Handlung heben sich von den meist üblichen Schießspielen ab: "Royol der Perverse", seines Zeichens Tyrann auf Saturn, bricht den jahrhundertealten Vertrag eines Vorfahren, der Terra regelmäßige Uranlieferungen zusicherte. Die Regierung der Erde beschließt daraufhin, sich den notwendigen Rohstoff mit Gewalt zu beschaffen, und schickt ihren besten Piloten auf die gefährliche Reise. Hat dieser sich durch die Spielanleitung gelesen, die zwar in Deutsch, aber sehr klein gedruckt ist, bricht er auf und sieht den ersten der vier Bildschirme vor sich: Eine Sternenkarte mit einigen Punkten, die Saturnmonde symbolisieren. Per Knopfdruck düst man zum Ziel und erkennt unter sich die Mondoberfläche (zweiter Bildschirm). Jetzt bewegt man den Joystick, um mit Hilfe eines Geigerzählers das Uranlager zu finden, und zerstört mit der Bordkanone eine aufgestörte Verteidigungsbasis, was selbst Anfängern keine Probleme bereitet. Dann gerät man in den Transporterraum (dritter Bildschirm), der angeblich in hochauflösender Grafik dargestellt ist, und beamt einen Roboter auf die Mondoberfläche, eine je nach Magnetfeldstärke des Mondes mehr oder weniger schwierige Aufgabe. Ärgerlich ist nur, daß gerade dieser an sich schwierigste Teil des Spiels auf einem winzigen Bildschirmausschnitt stattfindet, während ansonsten immer noch der relativ unwichtige Transporterraum gezeigt wird.



Am Ziel angekommen, sieht man den einzigen grafisch überzeugenden Bildschirm, das Uranfeld, wo der arme Roboter sich zum Uran stehlend und bis auf ein Schutzfeld unbewaffnet mit Laserbatterien und Müllbeseitigungssatelliten auseinandersetzen muß. Nach erfolgreicher Rückbeamaktion wird eindrucksvoll die Geschwindigkeit des C 64 demonstriert, denn der Uranzähler bewegt sich so schnell, daß es unmöglich ist, ihn abzulesen, und verschwindet dann wieder. Jetzt wiederholt sich der ganze Vorgang mit steigendem Schwierigkeitsgrad, d.h. stärkere Magnetfelder, mehr Laserbatterien usw. Obwohl die relativ friedliche Spielidee sowie die Verteilung der Handlung auf mehrere Schauplätze Beifall verdienen, ist das Spiel dennoch langweilig, weil die Spielaktionen entweder viel zu leicht oder viel zu schwierig sind und die echte "Action" eben fehlt.

System	C 64
Hersteller	Eurogold

Axel Klenk

Empire

Dieses Weltraumspiel unterscheidet sich von anderen Vertretern seiner Art durch eine einmalige Hintergrundhandlung. Der Spieler hat als Raumschiffpilot die Aufgabe, ein großes galaktisches Imperium aufzubauen. Er muß mit Sternenbasen aus Dutzenden von Sonnensystemen verhandeln, gegen Außerirdische kämpfen, Seuchen überleben und eine Vielzahl von Spezialmissionen bewältigen. Beim Flug durch die Galaxien steht ihm ein Bord-Computer (Meldung in Deutsch) zur Verfügung, der viele wichtige Daten anzeigt, wie z.B. den Energievorrat des Raumschiffs, die Temperatur, die Radioaktivität, die beim Befördern bestimmter Güter zum Strahlentod führen kann, den Schild-Status sowie die

Richtung zur nächsten Sternenbasis oder zu einem bestimmten Zielpunkt (Marker), den man in einer Karte selbst setzen kann. Beim Flug durch die Sonnensysteme zeigt der Bord-Computer außerdem die Richtung zum nächsten System. Ein Treibstoff-Meter gehört ebenfalls zur Ausrüstung des Piloten.

Für die Lösung der Aufgabe, die Gründung eines Imperiums, benötigt man nun Material, das auch im All nicht überall anzutreffen ist. Um sich dieses Material zu beschaffen, fliegt man durch ein Warp-Loch und gelangt schließlich auf einen Planeten im betreffenden Sonnensystem. Von dort wird man gemeinerweise mit Raketen beschossen. Ist man allen Raketen ausgewichen oder hat sie gar zerstört, kann man mit einigem Geschick Fracht (Mineralien, radioaktive Erze und Arbeiter) aufsammeln und zur zuständigen Sternenbasis befördern. Bei ausreichendem Angebot an Material kann sie Gegenstände produzieren, die zur Gründung des Imperiums benötigt werden. Manche Sternenbasen bieten Spezialmissionen an, die innerhalb eines gewissen Zeitlimits ausgeführt werden müssen. Auf der Reise durch das jeweilige Sonnensystem ist die System-Karte behilflich, auf dem "kleinen Trip" zwischen den Systemen ist die Galaxis-Karte nützlich. Als besonderen Gag bietet die System-Karte eine Zoom-Option, mit der man den jeweiligen Abschnitt, in dem man sich befindet, vergrößern kann.

"Empire" ist ein Spiel mit einem recht anspruchsvollen Schwierigkeitsgrad. Schade ist nur, daß keine High-Score-Liste angelegt werden kann und der Schwierigkeitsgrad nicht frei wählbar ist. Ein Score wird nicht angezeigt, am Ende des Spiels erfährt der Spieler lediglich, wieviel Prozent von "Empire" er erreicht hat. Die Grafik ist gelungen, dafür stört der schlechte Sound bereits nach fünf Minuten ganz gewaltig. Zum Glück läßt sich das dauernde Gedudel mit einem Druck auf die Funktionstasten abstellen. Positiv ist außerdem zu vermerken, daß sich der jeweilige Spielstand abspeichern läßt und das Spiel nicht wie andere Weltraumspiele (z.B. "Space Invaders") in sinnlose Schlachtereien ausartet. Die Meldungen im Spiel sind – wie bereits erwähnt – in Deutsch geschrieben. "Empire" ist somit ein durchaus interessantes Spiel mit einem etwas vom Standard abweichenden Hintergrund, gedanken.

System C 64
Hersteller Firebird Software

Stefan Schrack

Tips und Tricks zur Hardware des C 64 und der Floppy 1541

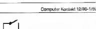
Diesmal stellen wir Ihnen drei Umbauten vor, die Ihnen die Arbeit mit Ihrem Computer erleichtern sollen. Bevor Sie jedoch zu Ihrem Lötkolben greifen und sich an die Arbeit machen, müssen folgende Punkte klargestellt werden:

1. Durch das Öffnen verlieren Sie Ihren Garantieanspruch.
2. Wenn Sie im Löten nicht geübt sind, proben Sie es erst einmal an einem Kabel!
3. Verwenden Sie kein Lötfett!
4. Überprüfen Sie jeden Arbeitsschritt genau!

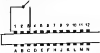
So, und jetzt kommen wir zu den Einbauanleitungen.

Einbau eines Spiele-Stop-Schalters

Mit diesem Schalter können Sie jedes Programm anhalten und zu einem anderen Zeitpunkt wieder fortsetzen. Benötigt werden zwei dünne Kabel und ein Schalter. Schalten Sie Ihren C64 aus und entfernen Sie alle Kabel (Netzkabel, Floppykabel usw.). Drehen Sie Ihren Computer um und lösen Sie die Schrauben der Unterseite. Jetzt können Sie das obere Gehäuseteil vorsichtig hochheben. Ziehen Sie den Stecker, der die Tastatur mit dem Computer verbindet (linke Platinenseite) und den Stecker für die LED (rechte Platinenseite) vorsichtig heraus, damit Sie das obere Gehäuseteil vollständig abnehmen können.



Stopschalter für Spiele



Reset Taster

In der rechten oberen Ecke der Platine sehen Sie nun den Modulport (Expansionsport). Hier löten Sie jeweils ein Kabel an Pin 1 (Masse) und Pin 4 (Interrupt) an (Bild 1). Vergewissern Sie sich, daß die Kabel sauber angelötet sind. Sie dürfen auf keinen Fall die Platine oder einen anderen Pin berühren. Danach müssen Sie nur noch die Kabel nach außen führen und durch einen Schalter verbinden.

Der Einbau eines Reset-Tasters

Ein Reset-Taster dient dazu, den Computer in den Ursprungszustand zu setzen, ohne jedoch den Speicher zu löschen (Kaltstart). Benötigt werden dazu zwei dünne Kabel und ein Taster. Entfernen Sie das Gehäuse Ihres Computers und ziehen Sie Tastatur- und LED-Stecker (wie in der ersten Einbauanleitung beschrieben). Auf der oberen linken Seite der Platine sehen Sie den Userport. Löten Sie jeweils an Pin 1 (Masse) und Pin 3 (Reset) ein Kabel an (Bild 2). Vergewissern Sie sich auch hier wieder, daß die Kabel sauber angelötet sind und führen Sie diese nach außen. Anschließend müssen Sie nur noch die Kabel mit einem Taster verbinden, und der Reset ist fertig.

Hardwaremäßiges Ausschalten der Lichtschranke in der Floppy

Wie Sie ja sicher wissen, kan man einseitige Disketten in de meisten Fällen auch beidseiti beschreiben, indem man ei zweites Loch an der linken Seit einzwickt. Dies wird überflüs sig, wenn man die Lichtschran ke der Floppy ausschaltet. Be nötigt werden dazu wieder zwe Kabel und ein Schalter.

Schalten Sie Ihre Floppy au und ziehen Sie den Netzstecker. Lösen Sie die Schrauben an de Unterseite der Floppy und öff nen Sie das Gehäuse. Auf de linken Platinenseite sehen Si nun mehrere Stecker. Uns in teressiert der Stecker mit de Bezeichnung P6. Um de Schreibschutz auszuschalte müssen Sie die beiden Kabe mit den Farben Orange und L la verbinden (Schalter). Vor sicht! Wenn der Schalter akti viert ist, haben auch di Schreibschutzaufkleber kein Funktion mehr und wichtig Daten können überschriebe oder gelöscht werden.

Gerold Herr

Ihr direkter Draht zur Redaktion.
Telefon 07252/42948
Für Briefe:
Computer Kontakt
Postfach 1648
7518 Bretten

Grafikkurs für den C 64

3. Teil

Heute wenden wir zum ersten Mal die bisher erworbenen Kenntnisse an, um zwei nützliche Routinen zu entwerfen, die Sie in Ihren eigenen Programmen problemlos verwenden können.

Anwendung 1: Screen-Switch

Die erste Routine besteht aus einem Maschinenprogramm, mit dem wir zwei Bildschirmseiten verwalten können, die voneinander völlig unabhängig sind. Als praktische Anwendung kann man sich zum Beispiel vorstellen, daß die erste Bildschirmseite dazu verwendet wird, um ein Hauptmenü darzustellen, und die zweite Seite, um die jeweils verschiedenen Unterprogramme anzuzeigen. Ist ein Unterprogramm beendet, muß das Hauptmenü nicht erst wieder von Grund auf neu konstruiert werden, sondern es wird einfach wieder die erste Bildschirmseite aktiviert. Besonders sinnvoll läßt sich die im folgenden vorgestellte Routine verwenden, wenn Informationen dargestellt werden sollen, deren Bildschirmaufbereitung normalerweise zuviel Zeit in Anspruch nimmt, so daß es sich nicht lohnt, dies immer wieder neu durchzuführen. In einem solchen Fall sollten die benötigten Informationen einmal dargestellt und dann im Bedarfsfall immer wieder ein- bzw. ausgeblendet werden können.

Welche Anforderungen müssen nun an die Routine zur Verwaltung zweier Bildschirme gestellt werden? Zunächst einmal ist klar, daß man zwei Speicherbereiche von jeweils 1000 Byte benötigt (25 Zeilen zu je 40 Bildschirmzellen), um zwei unabhängige Bildschirmseiten darstellen zu können. Zwischen diesen beiden Bereichen soll später in der Art eines Flip-Flop umgeschaltet werden. Weiterhin muß für jeden Bildschirm ein eigener Farbspeicher zur Verfügung stehen, damit jedes Zeichen auf dem Bildschirm in einer individuellen Farbe dargestellt werden kann. Wenn dann der jeweilige Bildschirm inaktiv geschaltet wird, muß der dem Bildschirm zugehörige Farbspeicher abgelegt werden, damit die Farbinformation nicht verlorengeht. Das wäre nämlich der Fall, wenn in den jeweils anderen Bildschirm geschaltet wird, da sich die beiden Bildschirme einen Farbspeicher teilen müssen. Und schließlich muß man sich, wenn von Bildschirm A auf Bildschirm B umgeschaltet wird, die aktuelle Cursorposition von Bildschirm A merken. Das ist sinnvoll, wenn man wieder zu Bildschirm A zurückkehren will, denn dann sollte sich der Cursor natürlich an der alten Stelle befinden.

Mit diesen Überlegungen können wir nun daran gehen, das Problem zu programmieren. Dabei empfiehlt sich der Einsatz von Maschinensprache, damit die Routine später universell einsetzbar ist. Außerdem lassen sich bestimmte Operationen leider nur in Maschinensprache durchführen, und auch die Geschwindigkeit des fertigen Maschinenspracheprogramms ist nicht zu unterschätzen. Wer sich in der Prozessorsprache des C 64 nicht so gut auskennt, der braucht an dieser Stelle nicht zu verzagen, denn ein Basic-Lader für die Initialisierung des eigentlichen Programms ist in Bild 1 zu finden.

Zunächst ein paar Worte bezüglich der verwendeten Adressen und der gewählten Speicherkonfiguration. Das Maschinenprogramm beginnt bei $9035 (=36917), der erste Bildschirm wird bei $8000, der zweite bei $8400 initialisiert. Der Farbspeicher für den ersten Bildschirm beginnt bei $9000 (=36864) und belegt 4*256 Byte, weswegen die zweite Bildschirmseite bei $9400 (=37888) beginnt. Wenn Sie nochmals in unseren letzten Kursteil hineinschauen, werden Sie sehen, daß die Adressen nicht zufällig so wie hier beschrieben gewählt wurden, sondern sich auf die Beispiele im zweiten Teil des Kurses beziehen. Nach dieser kurzen Einführung definieren wir nun in Assemblersprache einige Konstanten, auf die man später mittels des vergebenen Namens zurückgreifen kann. Das vereinfacht das Lesen des Listings wesentlich. Hier nun die Konstanten:

```
03 CIA_1     DE $DD00     Basisadresse des CIA 1 Beschreibung
                          siehe Grafikkurs Teil 2
04 VI_ZEICH  DE 53248+24  Register 24 des VIA Video-RAM &
                          Zeichengenerator
05 BILDZEIG  DE 648       Speicherzelle für Zeiger auf aktuellen
                          Bildschirmanfang Hi-Byte
06 SPALTE    DE 211       Zeiger auf aktuelle Spalte
07 ZEILE     DE 214       Zeiger auf aktuelle Zeile
08 SETCURS   DE 58732     Betriebssystemroutine setzt den Cursor
                          gemäß den Werten in ZEILE und
                          SPALTE
```

Die Zahlen ganz am Anfang jeder Zeile sind Zeilennummern, damit auf bestimmte Zeilen innerhalb dieses Kursteils leichter Bezug genommen werden kann.

Zunächst beschäftigen wir uns mit der Initialisierung, d.h. mit den notwendigen Operationen, die durchgeführt werden müssen, damit später die Umschaltung mittels eines Aufrufs durchgeführt werden kann. Hierzu dienen die folgenden Maschinensprachebefehle, die dasselbe wie die vorgestellte Basic-Routine unter der Überschrift "das Video-RAM" im zweiten Teil dieses Kurses bewirken:

```
10 INIT LDA CIA_1
11      AND #%11111100
12      ORA #%00000001   ;Block 1 aktiv schalten
13      STA CIA_1
14      LDA #%00000100   ;Zeichengenerator auf Offset
                         #4096 = $9000
15                       ;gleichzeitig Offset für Video-RAM
                         = $8000
16      STA VI_ZEICH
17      LDA #128
18      STA BILDZEIG     ;Bildschirmzeiger korrigieren
```

Mit dieser Befehlssequenz ist die eigentliche Initialisierung schon abgeschlossen, und die aktuelle Bildschirmseite beginnt nun bei $8000. Wie im vorhergehenden Kursteil konnten Sie nun durch POKE 32768,42: POKE 55296,1 in der linken oberen Ecke des Bildschirms einen weißen Stern erzeugen. Innerhalb der Initialisierungsroutine wird jetzt noch eine Cursorpositionstabelle eingerichtet, in der die aktuellen Cursorpositionen nach Zeile und Spalte festgehalten werden. Aus diesem Grund benötigt man für 2 Bildschirme 4 Byte, die jetzt belegt werden.

```
19               LDY #3
20               LDA #0
21 LOOP_001 STA CURSOR,Y  ;Cursorpositionstabelle initialisieren
22               DEY
23               BPL LOOP_001
24               RTS
```

Wenn man nun von der gerade aktivierten Bildschirmseite zur deaktivierten schalten will, muß man sich überlegen, welches Register zur Bestimmung des aktuellen Zustands herangezogen werden soll. Um das festzustellen, sind in Bild 1 die Inhalte der für die-

Register	symbolischer Name	Seite 1 $8000	Seite 2 $8400
#56576	CIA_1	XXXX XX01	XXXX XX01
#53272	VI_ZEICH	0000 010X	0001 010X
#648	BILDZEIG	# 128	# 132

Bild 1 Relevante Register und ihre Inhalte

se Routine relevanten Speicherzellen und deren Inhalt in Abhängigkeit von der aktivierten Bildschirmseite in einer Tabelle aufgeführt. Hier steht X für ein Bit, dessen Wert unbekannt ist. Die ersten beiden Angaben sind binär zu lesen, die letzte dezimal. Offenbar kann man zur Abfrage des aktuellen Zustands das Register CIA-1 oder aber das Register BILDZEIG verwenden. Der Einfachheit halber wird hier das Register BILDZEIG gewählt.

```
26 FLIP  LDA BILDZEIG     feststellen, ob im 1. oder 2. Bildschirm
27       AND #%00000100
28       PHA              und merken
```

Anhand von Bit 2 läßt sich eindeutig bestimmen, ob man sich zum Zeitpunkt des Aufrufs der Routine in dem ersten oder zweiten Bildschirm befand. Als nächste Aktion muß vor der Umschaltung auf die komplementäre Bildschirmseite die Cursorposition des noch aktiven Schirms gerettet werden.

```
29       LSR A            Accu enthält #0 oder #2
30       TAY              diesen Wert als Y-Index wählen
31       LDA *SPALTE      die aktuelle
32       STA CURSOR,Y     Cursorposition
33       LDA *ZEILE       in der Tabelle
34       INY              festhalten
35       STA CURSOR,Y
```

Danach ist es notwendig, das Farb-RAM des aktivierten Bildschirms zu sichern. Das geschieht mit einer Verschiebungsroutine, die diesen RAM-Bereich kopiert. Da diese Routine zweimal mit unterschiedlichen Werten aufgerufen wird, muß man angeben, wo der zu kopierende Speicherbereich beginnt und wohin dieser Speicherbereich kopiert werden soll.

```
36       LDA #$D8         Beginn des Farb-RAM
37       STA VON+1
38       PLA              gemerkten Wert abrufen
39       PHA              und weiter merken
40       ORA #$90         es wird entweder nach $9000 oder
41       STA NACH+1       aber nach $9400 kopiert
42       JSR SCHIEBEN     Verschiebung ausführen
```

Damit sind alle Operationen ausgeführt, die den zu desaktivierenden Bildschirm betreffen. Jetzt kann die Umschaltung und damit die Aktivierung des zweiten Schirms erfolgen.

```
43       LDA VI_ZEICH
44       AND #%00001111   Video-RAM vorbereiten
45       STA VI_ZEICH
46       PLA              Bildschirmanfang wiederholen
47       EOR #%00000100   und umschalten
48       PHA
49       ASL A            auf Bitposition 4 schieben
50       ASL A
51       ORA VI_ZEICH     neue Anfangsadresse
52       STA VI_ZEICH     des Video-RAM setzen
```

Damit ist an dieser Stelle die komplementäre Bildschirmseite aktiviert. Nun muß der Bildschirmzeiger entsprechend der Tabelle in Bild 1 korrigiert werden.

```
53       PLA
54       PHA
55       ORA #$80         High Byte des Bildschirm-Beginns
56       STA BILDZEIG     korrigieren
```

Abschließend sind noch zwei Dinge zu erledigen. Zum einen muß das Farb-RAM des zweiten Schirms restauriert werden, zum anderen ist ebenfalls der Cursor für diese Bildschirmseite zu positionieren. Hier die entsprechenden Maschinenspachebefehle in Assembler-Notation:

```
57       ORA #$90         Farb-RAM aus RAM
58       STA VON+1        holen und nach
59       LDA #$D8         $D800
60       STA NACH+1       kopieren
61       JSR SCHIEBEN
62       PLA              außerdem Cursorposition
63       LSR A            aus der Tabelle
64       TAY              holen
65       LDA CURSOR,Y     und in die Register
66       STA *SPALTE      für Zeile und
67       INY              Spalte schreiben
68       LDA CURSOR,Y
69       STA *ZEILE
70       JMP SETCURS      Cursor auf neue Position bringen
```

Mit dem letzten Befehl wird der Cursor gemäß den Werten fü Zeile und Spalte aus der Cursortabelle neu gesetzt. Damit ist de Flip-Flop-Umschalter ausgeführt. Der Vorteil dieser Routine lieg nun darin, daß bei einem nochmaligen Aufruf derselben Routin wieder in den ursprünglichen Bildschirm zurückgesprungen wird. Es bleibt das Unterprogramm SCHIEBEN nachzutragen, welch zweimal aufgerufen wird.

```
72 SCHIEBEN LDA #0          Blockverschiebung
73          STA *$22
74          STA *$58
75 VON      LDA #$XX        XX wird durch das rufende Programm
                            angegeben
76          STA *$23
77 NACH     LDA #$YY        YY wird durch das rufende Programm
                            angegeben
78          STA *$59
79 PAGES    LDX #$04        Anzahl der zu verschiebenden Seiten
80 LOOP_001 LDY #$00        Schleife
81 LOOP_002 LDA ($22),Y
82          STA ($58),Y
83          INY
84          BNE LOOP_002
85          INC *$23
86          INC *$59
87          DEX
88          BNE LOOP_001
89          RTS
```

Damit ist die Routine abgeschlossen. Der Vollständigkeit halbe wird nun noch die Tabelle für die Cursorposition definiert.

```
90 CURSOR   DS 4            4 Byte für Cursorpositionen reservieren
```

Das hier vorgestellte Assembler-Listing können Sie ohne Probleme mit einem eigenen Assembler neu eingeben und in Maschinensprache übersetzen, falls Sie einen solchen besitzen. Einen Basic-Lader für das eigentliche Maschinenprogramm finden Sie i Bild 2. Er ist selbsterklärend, weswegen hier auf das Programm nicht weiter eingegangen wird.

Ist das Maschinenprogramm durch das Basic-Programm erzeugt, wird die Routine INIT automatisch ausgeführt (SYS 38912). Sie kann im Bedarfsfall immer wieder aufgerufen werden. Die Umschaltung zwischen den beiden Bildschirmseiten erfolg durch Eingabe von SYS 38943. Um einen ersten Eindruck übe den Verwendungszweck für die Flip-Flop-Routine zu gewinne können Sie das Listing aus Bild 3 abtippen. Sie können auch eige Ideen im Umgang mit dieser Routine entwickeln.

Wir wenden uns nun einem weiteren, sehr interessanten The im Zusammenhang mit der Grafikprogrammierung zu: der Konstruktion von neuen Zeichen.

Anwendung 2: Zeichensatzprogrammierung

Haben Sie sich auch schon einmal darüber geärgert, daß Sie a Ihrem Bildschirm nicht die deutschen Umlaute darstellen könne Für eine gute Textverarbeitung bildet aber gerade der deutsch Zeichensatz die Grundlage. Der C 64 besitzt 2 Zeichensätze zu 256 Zeichen. Der erste umfaßt die Großbuchstaben und die Grafikzeichen, der zweite die Groß- und Kleinbuchstaben. Beide Zeichensätze sind vollständig im Anhang des C 64-Handbuchs aufgeführt (in der Tabelle der Bildschirmcodes). Die Umschaltung zwischen diesen beiden Zeichensätzen erfolgt normalerweise durch die gleichzeitige Betätigung der Commodore- und der Shift-Taste.

Bereits im ersten Teil dieses Kurses wurde beschrieben, wie ma ein dargestelltes Zeichen auf dem Bildschirm zusammensetzt. I Bild 4 wird am Beispiel des Zeichens A dieser Sachverhalt noc einmal gezeigt. Für jedes Zeichen steht grundsätzlich ein 8- Punkte-Feld zur Verfügung, innerhalb dessen man jeden einzeln

Bild 2: MC-Generator

```
1 PR=0:FOR I=38912 TO 39066.READ A:POKE I,A:PR=PR+A+1:NEXT
2 IF PR<>19156 THEN PRINT"PRUEFSUMMENFEHLER!":STOP
3 SYS 38912:POKE 56,128.POKE 55,0:CLR
4 PRINT"{CLR}SCREEN- SWITCHING IST AKTIVIERT!"
5 PRINT"{DOWN}UMSCHALTUNG MIT SYS 38943"
6 PRINT"{DOWN}INITIALISIERUNG MIT SYS 38912":END
7 DATA 173,000,221,041,252,009,001,141
8 DATA 000,221,169,004,141,024,208,169
9 DATA 128,141,136,002,160,008,169,000
10 DATA 153,155,152,136,016,250,096,173
11 DATA 136,002,041,004,072,074,168,165
12 DATA 211,153,155,152,165,214,200,153
13 DATA 155,152,169,216,141,129,152,104
14 DATA 072,009,144,141,133,152,032,122
15 DATA 152,173,024,208,041,015,141,024
16 DATA 208,104,073,004,072,010,010,013
17 DATA 024,208,141,024,208,104,072,009
18 DATA 128,141,136,002,009,144,141,129
19 DATA 152,169,216,141,137,152,032,122
20 DATA 152,104,074,168,185,155,152,133
21 DATA 211,200,185,155,152,133,214,076
22 DATA 108,229,169,000,133,034,133,088
23 DATA 169,208,133,035,169,224,133,089
24 DATA 162,004,160,000,177,034,145,088
25 DATA 200,208,247,230,035,230,089,202
26 DATA 208,240,096
```

Bild 3: Flip-Flop-Routine

```
10 IF PEEK(38912)<>173 AND PEEK(56)<>128 THEN PRINT"BITTE ERST INITIALIS
   IEREN!":STOP
11 POKE 53280,14:POKE 53281,6:SYS 38912
12 PRINT"{CLR,WHITE}DER{SPACE,GREY 3}ERSTE{SPACE,GREY 2}BILDSCHIRM IST A
   KTIV":GOSUB 24
13 SYS 38943:PRINT"{CLR,WHITE}JETZT{SPACE,LIG.BLUE}IST{SPACE,GREY 2}DER
   ZWEITE BILDSCHIRM AKTIV":GOSUB 24
14 FOR J=1 TO 5:SYS 38943:GOSUB 24:NEXT J
15 PRINT"{2DOWN,WHITE}GUTEN":PRINT"{2DOWN,LIG.BLUE}DAS IST BILDSCHIRM{SP
   ACE,WHITE}1":GOSUB 24
16 SYS 38943:PRINT"{2DOWN,LIG.BLUE,6SPACE}TAG'":PRINT"{2DOWN,WHITE}DAS I
   ST BILDSCHIRM{2SPACE,LIG.BLUE}2".GOSUB 24
17 FOR J=1 TO 5:SYS 38943:GOSUB 24:NEXT J
18 FOR I=1 TO 100:SYS 38943:NEXT:GOSUB 24:GOSUB 24
19 FOR I=32768 TO 32768+1000:POKE I,45:POKE I+22528,12:NEXT
20 SYS 38943
21 FOR I=33792 TO 33792+1000:POKE I,43:POKE I+21504,15:NEXT
22 FOR J=1 TO 5:SYS 38943:GOSUB 24:NEXT J
23 FOR I=1 TO 100:SYS 38943:NEXT:GOSUB 24:PRINT"{CLR}";:END
24 FOR I=1 TO 1000:NEXT:RETURN
```

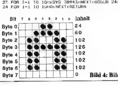


Bild 4: Bildschirmraster

Punkt setzen kann. Das gesamte Feld ist in 8 Byte zu je 8 Bit organisiert. Für ein gesetztes Bit muß dabei an der entsprechenden Stelle der Bitwert 1 geschrieben werden. Für das Zeichen A in unserem Beispiel sind die Inhalte der Bytes 0 bis 7 aufgeführt. Auf die Berechnungsweise wird an dieser Stelle nicht mehr gesondert eingegangen; im Anfangskapitel des Grafikkurses können sie gegebenenfalls Ihre Kenntnisse über die Art und Weise der Berechnung wieder auffrischen.

Um nun einzelne Zeichendaten zu ändern, muß die Basisadresse des Zeichengenerators bekannt sein, sie liegt bei $D000 (#53248). Und das bringt ein Problem mit sich. Weiter oben wurde die Basisadresse vom CIA 1 mit $D000 angegeben. Damit liegen der Input/Output-Baustein und der Zeichengenerator an derselben Adresse. Aus diesem Grund sind die Daten des Zeichengenerators normalerweise nicht lesbar. Trotzdem kann man aber mit Hilfe eines Interrupt-Bits in einem Kontrollregister vom CIA 2 den Zeichengenerator einblenden. Das werden wir später ausnutzen, um den Zeichengenerator im Rechner in einen neuen Speicherbereich zu verschieben. Da der originale Zeichengenerator selbstverständlich als ROM ausgeführt ist, muß man notwendigerweise diesen in einen Bereich des RAM kopieren, damit er dort verändert werden kann. Wie viele Daten sind aber zu kopieren?

Ein Zeichen besteht aus einem Block von 8 Byte. Es existieren zwei komplette Zeichensätze zu je 256 Byte, wie oben angeführt, so daß der Zeichengenerator 4 KByte belegt. Wenn diese 4 KByte kopiert sind, dann muß man selbstverständlich die Register des VIC II benutzen, um diesen Zeichensatz zu aktivieren. Natürlich kann der kopierte Zeichensatz durch entsprechende POKE-Befehle manipuliert werden. Mit Hilfe dieser Vorgehensweise werden schließlich auch die deutschen Umlaute entstehen. Vorher ist aber noch ein wenig Berechnungsarbeit erforderlich.

Zu Beginn ist zu überlegen, wo der neue Zeichensatz abgelegt werden soll. Da Zeichengenerator und Video-RAM im selben 16-K-Block untergebracht werden müssen, sind auch dafür Berechnungen notwendig. In unserem Beispiel legen wir das Video-RAM auf Adresse $4C00 und den Beginn des Zeichengenerators auf $5000. Damit dieser Bereich später nicht von einem Basic-Programm überschrieben werden kann, wird der obere RAM-Bereich gesperrt.

In Bild 5 sehen Sie die Zeichen, die in den neuen Zeichensatz implementiert werden sollen. Zusätzlich zu den deutschen Umlauten wurde als Sonderzeichen noch das wohlbekannte Copyright-Zeichen aufgenommen. Die dezimalen Werte der einzelnen Bytes eines Blocks sind ebenfalls aufgeführt. Jetzt muß man sich nur noch überlegen, welche Zeichen auf der Tastatur den neuen Zeichen weichen sollen. Da die in Bild 5 dargestellten Zeichen ihre Hauptanwendung in der Textverarbeitung finden, werden wir hier nur die Zeichen des Klein-/Großbuchstabensatzes verändern.

Wenn durch B die Basisadresse des kopierten Zeichengenerators bezeichnet wird, durch N die Nummer des Zeichensatzes (0 oder 1) und durch Z die laufende Nummer des Zeichens, das verändert werden soll (im Bereich 0 bis 255), dann bestimmt man die Basisadresse BA dieses Zeichens mit Hilfe der Formel

BA = B + N * 2048 + Z * 8

Ist die Basisadresse BA ermittelt, so muß man einschließlich dieser Speicherzelle die 8 Byte des Blocks ändern, der dieses Zeichen bildet. Damit ist unser "Handwerkszeug" schon komplett, und wir können damit beginnen, die eigentliche Routine zu entwerfen. Dazu muß zunächst der Zeichengenerator kopiert werden.

```
10 POKE 56,76: POKE 55,0: CLR: REM oberen Basic-Bereich
   sperren
11 POKE 56334,127: POKE 1,51: REM Zeichengenerator kann
   gelesen werden
12 FOR I=53248 TO 57343: REM 4 KByte des Zeichengenerators
   kopieren
13 POKE I-32768,PEEK (I)
14 NEXT I
15 POKE 1,55: POKE 56334,129: REM Zeichengenerator wieder
   ausblenden
```

Durch diese Zeilen wurde der CIA 1 zeitweilig ausgeblendet, damit ein Zugriff auf den Zeichengenerator erfolgen konnte. Nachdem sämtliche Zeichendaten kopiert sind, muß der ursprüngliche Zustand wieder hergestellt werden, da der CIA 1 ansonsten keine seiner für den normalen Betrieb notwendigen Aufgaben erfüllen kann. In den nächsten beiden Zeilen wird dann der 16-KByte-Block ausgewählt, in dem das Video-RAM und der Zeichengenerator liegen. Außerdem werden die notwendigen Informationen in das Register #24 des VIC II geschrieben, um den Beginn des Bildschirms auf $4C00 und die Basisadresse des zu manipulierenden Zeichensatzes auf $5000 zu legen. Zusätzlich muß in das Register BILDZEIG der Wert $4C (#76) geschrieben werden.

```
16 POKE 56576,(PEEK(56576) AND 252) OR 2: REM 16-
   KByte-Block wählen
17 POKE 53272,48+4: POKE 648,76: REM Zeichengenerator
   und Video-RAM
```

Jetzt können die neuen Zeichendaten an die entsprechenden Stellen geschrieben werden, um auf diese Art und Weise einen neuen Zeichensatz zu erhalten. Die Tabellenübersicht in Bild 6 verdeutlicht den Zusammenhang zwischen dem neuen und dem alten Zeichensatz.

Durch unsere Manipulationen werden natürlich Zeichen des ursprünglichen Schriftsatzes gelöscht. Die in der Tabelle dargestellte Belegung wurde so gewählt, daß die deutsche Normbelegung für diese Tasten aktiviert wird. In unserem Programm können die Zeichendaten für den Zeichensatz am einfachsten über DATA-Zeilen eingelesen werden. Dabei wird vorausgesetzt, daß die Basisadresse BA des jeweiligen Zeichens bekannt ist. Aus diesem Grund beginnt auch jeder Datensatz für ein Zeichen mit der Basisadresse BA.

```
18 FOR J=0 TO 7: REM 8 Zeichen zu bearbeiten
19 READ BA: FOR I=BA TO BA+7: REM 8 Zeichendaten zu
   setzen
20 READ Z: POKE I,Z
21 NEXT I: NEXT J: END
```

Abschließend nun noch die Daten, die sich auf die Angaben in Bild 5 stützen:

```
22 DATA 22528,0,102,0,102,102,102,62,0
23 DATA 22744,102,0,60,102,102,102,60,0
24 DATA 22752,124,102,124,102,124,96,96,0
25 DATA 22760,195,60,102,126,102,102,102,0
26 DATA 22792,0,102,0,60,102,102,60,0
27 DATA 23000,102,0,60,6,62,102,62,0
28 DATA 23368,60,66,153,161,161,153,66,60
29 DATA 23504,102,0,102,102,102,102,60,0
```

Die Zeilen 22 bis 29 bestehen jeweils aus der Basisadresse BA des Zeichens und den 8 Zeichendaten, die das neue Zeichen formen. Wenn Sie das Programm ordnungsgemäß eingetippt haben, müssen Sie nach dem Start des Programms mittels der Commodore- und der Shift-Taste den zweiten Schriftsatz aktivieren. Danach erscheinen die neuen Zeichen bei Betätigung der entsprechenden Taste.

Wenn Sie die Routine verfeinern wollen, dann können Sie die Zeichendaten so ergänzen, daß die Tastenbelegung der DIN-Tastatur entspricht. Dazu müssen nur noch die beiden Zeichen "<" und ">" durch die Zeichen "" und "" ersetzt werden. Außerdem könnte man die Tabelle noch so ergänzen, daß die neuen Schriftzeichen invers darstellbar sind. Wenn Sie einmal mit dem Cursor auf ein neu definiertes Zeichen fahren, wird dieses Zeichen abwechselnd normal und invers dargestellt. Da aber zunächst die Definitionen für den inversen Schriftsatz noch nicht vorhanden sind wird als inverses Zeichen das der ursprünglichen Tastenbelegung entsprechende inverse Zeichen angezeigt. Wenn Sie also unsere DATA-Tabelle um die inversen Zeichen ergänzen, können Sie den gesamten deutschen Schriftsatz auch invers darstellen!

So, damit ist der dritte Teil des Kurses beendet. Im nächsten und letzten Teil des Grafikkurses werden wir uns dann mit den verschiedenen Grafikmodi beschäftigen, die der C 64 zur Verfügung stellt. Bis dahin wünsche ich Ihnen jedenfalls viel Spaß beim Ausprobieren der diesmal vorgestellten Anwendungen.

Frank Lasowski

Bild 5: Raster der neuen Zeichen

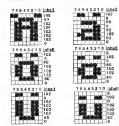


Bild 6: Alter und neuer Zeichensatz

Zeichennummer	altes Zeichen	neues Zeichen	Basisadresse BA des Zeichens
0	@	ü	22528
58	:	ö	22992
59	;	ä	23000
122	┘	Ü	23504
27	[	Ö	22744
29	]	Ä	22760
28	£	ß	22752
105	◪	©	23368

Vietnam für den C 64

Ziel des Spiels ist es, möglichst weit in gegnerisches Gebiet einzudringen, ohne den pausenlos angreifenden Kontrahenten zum Opfer zu fallen. Zunächst werden Maschinenprogramme und Grafik eingelesen, was ca. 4 Minuten dauert. Sobald Musik ertönt, können Sie Ihre Spielfigur mit dem Joystick (Port 2) bewegen. Sie sind natürlich nicht wehrlos und können durch Druck auf den Feuerknopf die feindlichen UFOs beschießen. Bei einem Treffer kassieren Sie Punkte. Sollten Sie selbst getroffen werden, so erscheint eine Auswertung, aus der ersichtlich wird, wie viele Meter Sie in Feindesland vorgedrungen sind. Nun können Sie durch Druck auf die Space-Taste ein neues Spiel beginnen.

Sollten Sie sich einmal in einer ausweglosen Situation befinden, so gelangen Sie mit der RUN-STOP/RESTORE-Taste wieder zum Titelbild. Tippen Sie dann GOTO 1300 auch ohne Cursor blind ein und nach Return beginnt das Spiel von neuem.

Christian Rabach

MC-Generator

```
0 POKE56,92
1 PRINT"█████V I E T N A M"
2 PRINT"█████(C) 89  RADDISH-SOFT GERMAN
Y"
3 PRINT"█████PLEASE WAIT 4 MINUTES"
9 FORI=900TO912:READQ:POKEI,Q:NEXT
10 DATA165,1,41,254,133,1,76,240,88,76,1
04,89,255
100 FORT=51200TO51967:POKET,0:NEXT
110 FORT=0TO32:READQ:POKET+51200,Q:POKE5
1264+T,Q:POKE51328+T,Q:POKE51392+T,Q
120 NEXT
125 FORT=0TO32:READQ:POKE51456+T,Q:POKE5
1520+T,Q:POKE51584+T,Q:NEXT
130 FORT=0TO32:READQ:POKE51648+T,Q:POKE5
1712+T,Q:POKE51776+T,Q:NEXT
135 FORI=0TO9:FORT=0TO17:READQ:POKE51233
+T+I*64,Q:NEXT:NEXT
136 FORI=1TO22:READQ:POKE51840+I,Q:NEXT
139 DATA0,2,0,0,10,128,0
140 DATA10,128,0,14,192,0,2,0,0,10,128,0
,10,128,0,10,128,0,10,128,0,15,192,0,10
146 DATA128,0,2,0,0,10,128,0,5,128,0,1,1
28,0,2,0,0,10,128,3,250,128,0,123,128
147 DATA0,46,128,0,15,192,0,10,128,0,2,0
,0,10,128,0,9,44,0,9,0,0,2,0,0,10,128,0
149 DATA10,191,0,11,180,0,10,224,0,15,19
2,0,10,128,0
150 DATA10,128,0,8,192,0,8,192,0,12,192,
0,12,0,0,12,0,0,10,128,0,8,128,0,8,192,0
151 DATA12,192
152 DATA0,12,192,0,12,0,0,10,128,0,8,128
,0,12,128,0,12,192,0,12,192,0,0,192,0
154 DATA10,128,0,12,128,0,12,128,0,12,19
2,0,0,192,0,0,192,0,10,128,0,40,128,0,32
156 DATA160,0,48,60,0,48,12,0,240,48,0,1
0,128,0,8,128,0,8,128,0,12,192,0,12,192
158 DATA0,63,0,0,8,128,0,10,128,0,2,128,
0,0,192,0,3,192,0,3,192,0,10,128,0,8,160
160 DATA0,40,32,0,240,48,0,192,48,0,48,6
0,0,10,128,0,8,128,0,8,128,0,12,192,0,12
162 DATA192,0,3,240,0,8,128,0,10,128,0,1
0,0,0,12,0,0,15,0,0,15,0
163 DATA240,0,6,89,0,21,165,64,80,85,80,
238,238,224,85,85,80,21,80,44,1,244
170 POKE53270,216
```

```
172 POKE53282,5:POKE53283,11
180 FORX=832TO863:READQ:POKEX,Q:NEXT
190 DATA120,169,51,133,1,169,0,133,95,13
3,90,133,86,169,208,133,96,169,240,133
200 DATA89,169,224,133,91,32,191,163,169
,55,133,1,88,96
210 SYS832:POKE650,160:POKE56576,PEEK(56
576)AND252:POKE53272,8:POKE648,192
220 B=57344:PRINT"L";CHR$(14)
230 READZE:IFZE=-1THEN1000
240 FORT=0TO7:READQ:POKEB+ZE*8+T,Q:NEXT
250 GOTO230
300 DATA27,0,,192,0,0,68,0,0,28,12,0,0,0
,48,0,0,0,29,0,3,0,0,0,0,192,0,30,85,85
310 DATA149,80,85,85,86,85,31,80,85,85,8
5,149,85,85,85,33,0,0,48,236,235,60,0,0
320 DATA30,0,3,14,58,234,234,63,195,36,0
,192,176,188,176,192,48,0,37,0,0,0,0,0
330 DATA15,58,59,58,0,0,0,48,234,187,170
,170,39,0,0,0,0,0,0,192,192,40,234,234
340 DATA238,230,255,58,62,15,41,184,171,
234,170,174,170,235,238,42,176,176,176
350 DATA176,240,240,240,204
360 DATA43,0,15,30,207,226,200,226,63,44
,0,235,34,255,34,136,34,255,45,0,252,47
370 DATA243,59,179,44,240,46,234,170,17,
251,174,171,171,254,47,186,174,174,255
380 DATA170,170,170,255,48,170,170,170,2
55,186,186,191,234,49,186,186,238,250
390 DATA170,170,171,254,50,171,171,171,1
95,233,234,234,238,51,0,0,0,0,0,15,58,63
400 DATA52,0,0,0,0,0,255,174,250,53,0,0,
0,0,0,255,174,255,54,0,0,0,0,0,255,170
410 DATA255,55,0,0,0,0,0,240,240,176,56,
0,3,3,0,3,3,3,0,57,176,176,176,176
420 DATA176,176,192,0,58,0,0,3,13,53,59,
58,58,59,0,15,245,21,81,85,245,175
430 DATA60,0,255,85,20,85,21,71,254,61,0
,193,124,87,85,30,250,170,62,0,0,0,0,192
440 DATA192,192,192,65,58,58,58,58,58,25
0,234,224,68,170,234,170,170,170,174,174
450 DATA171,65,170,234,171,171,170,187,1
70,170,66,171,171,171,171,171,170,234
460 DATA234,67,0,,0,0,0,192,193,176
470 DATA68,0,63,53,213,85,85,85,85,69,20
4,119,117,85,85,85,85,85,70,48,3,207,128
480 DATA85,85,101,85,71,0,0,207,117,85,8
5,85,85,72,85,85,85,95,87,223,51,0
490 DATA73,85,85,117,85,119,208,15,0,74,
117,85,245,117,85,127,192,0
500 DATA75,3,0,195,195,195,195,195,195,7
6,0,0,0,12,12,12,12,12,77,0,195,13,61,53
510 DATA53,53,53,78,85,87,87,87,87,151,8
7,87,79,195,12,12,12,12,12,12,12,80,12
520 DATA48,48,48,48,48,48,48,81,53,213,2
13,217,213,213,213,213,83,87,87,87,158
530 DATA92,92,92,92,83,12,12,12,48,48,48
,48,48,84,48,48,48,192,192,192,192,192
540 DATA80,213,213,213,213,229,213,213,2
13,86,92,92,240,112,115,112,192,0
550 DATA87,60,48,48,48,204,192,195,0,88,
192,192,204,192,193,204,195,0,89,213
560 DATA213,61,53,53,15,3,0
570 DATA90,0,0,0,0,0,0,3,3,91,0,3,15,63,
251,233,235,233,93,0,0,192,243,195,174
580 DATA170,170,93,0,0,0,0,192,240,192,1
74,94,0,0,0,0,0,60,239,171,95,0,0,0,0,0
590 DATA15,250,170,96,0,3,3,3,207,254,17
0,170,97,192,240,188,172,175,171,170,170
600 DATA98,0,0,0,3,15,234,170,170,99,0,0
,204,255,187,187,235,239,100,0,0,0,3,15
610 DATA15,14,14,101,15,14,62,250,234,17
0,170,170
620 DATA102,171,171,171,171,171,172,172,
176,103,170,170,170,170,255,0,0,0,104
630 DATA171,171,175,190,250,58,58,58,105
,255,171,171,171,171,171,172,172,106
640 DATA40,0,0,3,62,254,250,254,107,0,0,4
8,255,234,170,170,170,108,14,14,14,250
650 DATA250,170,170,170,109,170,170,171,
172,172,176,192,192,110,192,192,0,0,0,0
660 DATA0,0,111,58,58,58,58,59,59,60,60,
112,176,176,192,192,0,0,0,0,113,239,239
670 DATA234,234,234,234,234,234,114,170,
234,255,191,170,170,170,170,115,171,172
680 DATA172,252,255,171,170,170,116,0,0,
12,63,251,234,170,170,117,0,0,0,0,204
690 DATA255,187,170,118,0,0,0,0,0,0,192,
240,119,0,0,0,0,15,15,62,62,120,0,0,0
700 DATA48,255,239,170,170,121,0,3,3,255
,254,234,234,234,122,236,236,172,172
710 DATA172,172,172,172,123,48,48,0,0,0,
0,0,0,124,234,234,234,234,234,234,234
720 DATA234,125,170,170,170,170,170,170,
170,170,126,85,85,85,85,85,85,85,85
730 DATA127,190,175,171,170,170,170,170,
170,128,190,255,250,170,170,170,170,170
740 DATA129,250,234,170,170,170,170,170,
170,130,234,234,234,234,234,234,235,235
750 DATA131,172,172,176,176,192,192,0,0,
132,234,234,255,0,0,0,0,0,135,170,170
760 DATA255,0,0,0,0,0,134,234,234,240,0,
0,0,0,0
770 DATA135,15,13,3,0,0,0,0,0,136,0,195,
124,215,53,13,3,253,137,0,3,13,53,215
780 DATA93,95,80,138,68,215,127,192,0,0,
0,195,139,3,13,55,220,240,0,0,0
790 DATA140,85,253,0,0,0,0,0,0,141,85,87
,93,240,240,240,240,240,142,112,92,215
800 DATA53,13,3,0,0,143,0,0,0,192,112,11
2,195,0,148,240,240,240,240,240,240,240
810 DATA240,145,0,0,0,0,0,3,3,0,146,240,
240,240,252,252,252,255,0
820 DATA147,0,0,0,0,0,4,1,17,148,0,1,21,
4,63,17,21,65,149,0,0,0,16,20,64,4,21
830 DATA150,0,0,1,0,4,17,5,4,151,85,81,6
9,69,65,17,20,5,152,5,68,85,65,81,69,1
840 DATA21,153,5,69,21,84,5,85,21,65,154
,64,0,0,16,64,0,68,84,155,21,81,69,21
850 DATA16,20,0,0,156,84,65,17,69,64,64,
65,1,157,68,68,68,64,4,4,65,0
860 DATA158,64,69,85,65,84,16,0,0,159,20
,85,80,20,20,0,0,0
870 DATA160,0,63,51,51,51,51,51,63,161,0
,3,15,3,3,3,3,3,162,0,63,3,3,63,48,48
880 DATA63,163,0,63,3,3,63,3,3,63,164,0,
51,51,51,63,3,3,3,165,0,63,48,48,63,3,3
890 DATA63,166,0,63,48,48,63,51,51,63,16
7,0,63,3,3,12,12,48,48,168,0,63,51,51
900 DATA63,51,51,63,169,0,63,51,51,63,3,
3,63
999 DATA-1
1000 AD=22000:FORI=1000TO1135:READA$,PS:
C=0:FORN=1TOLEN(A$)STEP2
1001 A=ASC(MID$(A$,N,1))-48:B=ASC(MID$(A
$,N+1,1))-48:A=A-7*(A>9):B=B-7*(B>9)
1002 POKEAD,16*A+B:C=C+PEEK(AD):AD=AD+1
1003 NEXT:PRINTI;" ";:IFC<>PSTHENPRINT"E
RROR":STOP
1004 PRINT"OK":NEXT
```

```
1005 GOTO1200
1006 DATA A9008D8563BD8663BD8763BD8863BD
8963BD8A63BD8B63BD8C6349008DAC63, 3762
1007 DATA 8DAD638D8E638DAF638D71648D7264
8D73648D74648D75648A9018DAB638D00, 3763
1008 DATA A908990008990009A920990DC09900
C18AD0E8D0A90F8D18D4A96085F8A9608D, 4010
1009 DATA FCA250A000A804DC297FF008C901FD
0EA92091FB4C7756A92191FB4C7756A9, 3852
1010 DATA 2391FBC8A92491FB4C7756C800D8E6
FCC800D1A95F85FDA96185FEA00020BD, 4922
1011 DATA 36B1FDF07BC901F017C902F016C903
F015C904F014C905F013C906F0124C0E, 3759
1012 DATA 374C64574C8A574C1D584C66584C8C
584C125918A5FD690185FDA5FE690085, 3271
1013 DATA FE60A9A085FDA96085FC18A5FB6928
85FBA5FC690085FCCAD0F060A96085FB, 4933
1014 DATA A99585FC18A5FB692885FBA5FC6900
85FCCAD0F0B018A5FB692885FBA5FC69, 4816
1015 DATA 0085FC604C8D0320BD56B1FDF007AA
20C8564C255720BD56B1FDAA20E45620, 3661
1016 DATA 8D56B1FD8D890218A5FB6D890285FB
A5FC690085FCA94F858CA96185BDA203, 4382
1017 DATA A003B1BC91FB88D0F920FD5618A58C
690585BCA5BD690085BDC8D0E44C8756, 4414
1018 DATA 20BD56B1FDF007AA20C8564C785720
8D56B1FDAA20E45620BD56B1FD8D8902, 4079
1019 DATA 18A5FB6D890285FBA5FC690085FCA9
A985BCA96085BDA20AA003B1BC91FB88, 4511
1020 DATA D0F920FD5618A58C690585BCA5BD69
0085BDCAD0E44C875620BD56B1FDF007, 4344
1021 DATA AA20C8564CD15720BD56B1FDAA20E4
5620BD56B1FD8D890218A5FB6D890285, 4050
1022 DATA FBA5FC690085FCA9F785BCA96085BD
A204A003B1BC91FB88D0F920FD5618A5, 4726
1023 DATA 8C690485BCA5BD690085BDCAD0E44C
875620BD56B1FDF007AA20C8564C2758, 4012
1024 DATA 20BD56B1FDAA20E45620BD56B1FD8D
890218A5FB6D890285FBA5FC690085FC, 4348
1025 DATA A90785BCA96185BDA202A003B1BC91
FB88D0F920FD5618A58C690585BCA5BD, 4301
1026 DATA 690085BDCAD0E44C875620BD56B1FD
F007AA20C8564C7D5820BD56B1FDAA20, 4565
1027 DATA E45620BD56B1FD8D890218A5FB6D89
0285FBA5FC690085FCA91185BCA96185, 4312
1028 DATA BDA203A003B1BC91FB88D0F920FD56
18A58C690585BCA5BD690085BDCAD0E4, 4435
1029 DATA 4C875620BD56B1FDF007AA20C8564C
B35820BD56B1FDAA20E45620BD56B1FD, 4137
1030 DATA 8D890218A5FB6D890285FBA5FC6900
85FCA93385BCA96185BDA202A003B1BC, 4178
1031 DATA 91FB88D0F920FD5618A58C690585BC
A5BD690085BDCAD0E44C875620BD56B1, 4366
1032 DATA FDF007AA20C8564C295920BD56B1FD
AA20E45620BD56B1FD8D890218A5FB6D, 4064
1033 DATA 890285FBA5FC690085FCA93B85BCA9
6185BDA204A005B1BC91FB88D0F920FD, 4380
1034 DATA 5618A58C690385BCA5BD690085BDCA
D0E44C875678A97F8D0DDCA9018D1AD0, 3943
1035 DATA A9028502A97F8D0FDDA9B98DFCADA460
8D12D0A9188D11D0A9788D1403A9598D, 3964
1036 DATA 1503584C95594C5E598D17D08519D0
2901F0F3C602100489018D02A6028D86, 3040
1037 DATA 808D21D08DA4608D12D0E000F0034C
E059A011B9706399FFCF88D0F7A9FF8D, 4494
1038 DATA 1D0CA009B9B86399F7C388D0F74C53
58A011B9B26399FFCF88D0F7A9008D1D, 4449
1039 DATA 0CA009B9B26399F7C388D0F74C535B
AD00DC2910F019AD00DC2908F049AD00, 3918
1040 DATA DC2904F079AD00DC2901F001604CD3
3AADC763D00160A9008DC763AD8363BD, 3687
1041 DATA 8963AD8463AD8A63AD00DC2904F00D
AD00DC2908F00CA9008DC86360A9D188, 3419
1042 DATA C86360A9D28DC86360AD8363C9FA90
0160AD8363E9184A4A4AA8993C1C924, 4009
1043 DATA 9D0160EE8363AD8702C9038009EE87
02A9028D860260A9008D8702A902AD86, 3460
1044 DATA 0260AD8363C919B0018DAD8363E918
4A4A4AA8990C1C924900160CE8363AD, 3609
1045 DATA 8702C9039009EE8702A9018D860260
A9008D8702A9018D860260AD8363E918, 3334
1046 DATA 4A4A4A4A8B942C1C924900160AD8702
C9058006EE87024CE65AA9008D8702A9, 3541
1047 DATA 008D8602EE8663EE8963EE8A63EE8C
63AD8C63C97D9008A9018DC763A9008D, 4034
1048 DATA 8C63EAADA960C917F007EEA98D2032
60602032BD18A5FBE92785FBA5FCE900, 4083
1049 DATA 85FCA91D8DA9604C2F5BA90085FDA9
C085FEA000A5FB85BCA5FC85BDB1BC91, 4639
1050 DATA FD88D0F7E6BDE6FEA5FEC9C2D0EFAD
A6028DA86DAD1100AAF0066A866AAA8, 5127
1051 DATA AD9CDE5EADC763F00160ADC863F02C
C901F014AD8963C9FD9006A9018DC762, 4104
1052 DATA 6DEE8963EE896360AD8963C914B006
A9018DC76360CE8963CE896360AD8A63, 4047
1053 DATA C91EB006A9018DC76360CE8A63CE8A
6360ADC963F019A9008DC963AD04DC29, 3886
1054 DATA 03ADCA63AD8563BD8863AD8663BDAC
63ADCA63F02CC901F0144D8863C9FD90, 4244
1055 DATA 06A9018DC96360EE8863EE886360AD
8863C914B006A9018DC96360CE8863CE, 3932
1056 DATA 886360AD8C63C97F9006A9018DC963
60EE8C63EE8C6360201F5C20545C2088, 3813
1057 DATA 5C6DCE8902AD8502F00160A90A8D85
02AD8602F0DEC901F0148CB702899463, 3635
1058 DATA 8D836360ACB702899C638D836360AC
B702899B638D836360CEB202AD8202F0, 4114
1059 DATA 0160A90A8D8202AD8302F00EC901F0
14ACB402B9A2638D846360ACB402B902, 3684
1060 DATA 638DB46360ACB402B9A2638DA6363D
CEAF02ADAF02F00160A90A8DAF02A08D, 3847
1061 DATA 02C901F0A2C902F014ACB10289A263
8D56360ACB10289A2638D56360ACB1, 4046
1062 DATA 0289A2638D8636020C73C20D25D60
ADC463F037C901F007C902F0754C755D, 3777
1063 DATA AD8663C9B09006A9008DC4636020A1
5DAD52B1A7C924900160AD8402C90590, 3841
1064 DATA D549FF8D8402EE8402EE866360AD04
DC29038DC463E9018D8302AD8302F024, 3802
1065 DATA C902F010A9FE8D8563AD04DC293F69
348D866360A9188D8563AD04DC293F69, 3557
1066 DATA 348D866360A9348D8663AD04DC297F
69508D856360AD8563C91C8006A9008D, 3462
1067 DATA CA636020A15DA003B1A7C92490018D
AD8402C9039005A9FF8D8402EE8402CE, 3740
1068 DATA 8563808D8563C9FE9006A9008DC463
6020A15DA003B1A7C924900180AD8402, 3655
1069 DATA C9039005A9FF8D8402EE8402EE8563
60A9C085A8A90085A7AD8563E9184A4A, 4027
1070 DATA 4A85A7AD8663E9334A4A4AAAE000F0
1118A5A7692885A7A5A8690085A8CA4C, 3758
1071 DATA 8C5D60ADC563F037C901F007C902F0
764C813EAD8863C9B09006A9008DC563, 3943
1072 DATA AD20AD5EA052B1A7C924900160AD81
02C9059005A9FF8D8102EE8102EE8863, 3800
1073 DATA 8DAD04DC29038DC8638D8002C901F0
28C900F0EDC902F01DA9188D87634DD4, 3659
1074 DATA DC293F69348D8B6360A9FE8D8763AD
DADC293F69348D8B636DA9348D8B638D, 3651
```

```
1075 DATA 04DC297F69508DB76360ADB763C91C
B00BA9008DC5636020AD5EA000B1A7C9, 3572
1076 DATA 24900160BDB102C9039005A9FF8DB1
02EEB102CEB76360ADB763C9FE9006A9, 3860
1077 DATA D08DC5636020AD26A003B1A7C92490
0160ADB102C9039005A9FF8DB102EEB1, 3681
1078 DATA 02EEB76360A9C085A8A90085A7ADB7
63E91B4A4A4A85A7ADB863E9334A4A4A, 3801
1079 DATA AAE0D0F01118A5A7692885A7A5A869
0085A8CA4CC85E6020FE5920158C20CD, 3803
1080 DATA 5C20675B20825B20506020FC5E2032
5F20735F4C31EAAD8663C97D900AA900, 3033
1081 DATA 8DC463AD8863C97D9005A9008DC563
AD8A63C97D900AA97D8D8A63A9018DC7, 3997
1082 DATA 63AD8C63C97D9005A9018DC96360AD
1ED08DAA632908D008ADAA632901D022, 3305
1083 DATA 60ADAA632902D008ADAA632904D008
60A9008DC46320146060A9008DC56320, 3065
1084 DATA 146060ADAA632908F00160A9008DAB
6360ADAB63F0016078208C5FA0018C29, 3318
1085 DATA 10D0F9A9018DAB63408403A920A000
9900C09900C19900D89900D988D0F1A0, 3801
1086 DATA 1AB9CB639930C088D0F7A016B9E663
99E0C088D0F7A013B9FD6399D6C088D0, 4876
1087 DATA F7A005B9706A69A099D0C088D0F5A0
7264F00EA016B911649920C188D0F7AC, 4386
1088 DATA EB3FA013B92B649920C188D0F7A01F
B93C649978C188D0E7A004B9AB6369A0, 4312
1089 DATA 99ACC188D0F5A014B95C6499E8C188
D0F7A9D08D18D460EEAF63A003B9AC63, 4606
1090 DATA C90AD00EA90099AC6318B9AB636901
99AB6388D0EB60EE7564A00AB97164C9, 3924
1091 DATA 0AD00EA9009971641 8B97064690199
706488D0EB60CEB163F00160A90A8DB1, 3647
1092 DATA 63EE8063AD8063C940D005A9008D80
63AC8063A95A8D05D48D06D48D0CD48D, 4002
1093 DATA 0DD4A9208D04D48D0BD4A9218D04D4
8D0BD4B908528D01D4B948528D00D4B9, 3576
1094 DATA 89528D08D4B9C8528D07D4B0288F09
0910101B1C1B5A5B5C5D5E5F60616263, 2762
1095 DATA 1B1D44656667676767676768696A6B
6C6D6E1D1C1C1B1D1C6F707172737475, 2643
1096 DATA 761B777B0B7A7B1D7C7D7D7D7D7F80
812082831C1D84858585858585858586, 3660
1097 DATA 1C1D1C4A4B4C4D4E4F505152535455
565758593A3B3C3D3E3F404142432033, 2093
1098 DATA 34363436333436343637382E2F3031
322E2F30313234382E2F3031322E2F30, 1580
1099 DATA 31323925262728292A8788898A208B
8C8D8E8F209020202020919220202093, 2814
1100 DATA 949520969798999A9B9C9D9E9F0007
1E0E072A1707400307AC2207622107BF, 2477
1101 DATA 0807B01A07EB0C07F60807000D2207
00180207001A1207002A0207002A1F07, 1003
1102 DATA 002B1007003B1807003E0A07004220
07004D130700541F07008404070 08F08, 847
1103 DATA 070080170700D4040700EB1F012D04
015719012B1701798C01BA1101C50201, 1594
1104 DATA 081A01DA0101F4120100220E01005B
0701009A030100AF1501D0E40C031608, 1502
1105 DATA 03221F033601034C0C034D13036216
03791E03D30203E211022B2203EA0703, 1871
1106 DATA ED1003F30E03FC0803002B0B03004C
1B0300840102009A150300B0100300D3, 1665
1107 DATA 030300EC080300F321040203040719
040E0904160D04230404400B044C1304, 868
1108 DATA 5B01DA711B04B60004A71504DA01DA
000D110400390B040053010400600F04, 1109
1109 DATA 006A160400670504006B1404006F00
04007D1A0400B4110400BA0B0400BB19, 1109
1110 DATA 0400A5030400B31A0400E1050400EF
17000320050606051706082000B02719, 1121
1111 DATA 003514054415054002054E18055612
053E1F056F0305710805862005980705, 1179
1112 DATA A81E05B01005C20205D31905E71505
ED0B05F70205000C040500121F050011, 1695
1113 DATA 150500270E05003B0B0500DA1F0500
4A080500671305007903050086020500, 734
1114 DATA 911705009A1F0500A7220500AE0100
00B8170500C20C0500C81F0500CD0E05, 1635
1115 DATA 00D6180500DD040500E31F060D2006
2202064B1F065816065C0E0667200671, 1422
1116 DATA 04067B01067C0B067D11068B15069A
D906820206D51906F20C0600200C0600, 1497
1117 DATA 591506009A200600D1180600D60600
001B47489778974B97D89705972D9700, 2474
1118 DATA 47E0009767B3C28000000000000000
00000000002728292824252625202122, 1182
1119 DATA 2322212A2A2A2A2A2A2A2A00000000
00000000001010102E2E2E2E10002D2D, 776
1120 DATA 2D2D2D2D2D2D00010001000100200C
05090405122008011420051320130905, 580
1121 DATA 200512170913030814201309052013
090E04200102051220090D0D05120809, 445
1122 DATA 0E200D051405122017050914200705
0B0F0D0D050E20040113201701122013, 492
1123 DATA 03080F0E2007010E1A200715142004
01122017011220170F080C200E0F0308, 484
1124 DATA 14132001151313051204050D200801
02050E2013090520202020202010150E, 546
1125 DATA 0B1405201310010305201401131405
2004121505030B050E00000000000005, 327
1200 AD=21000:FORI=1206TO1214:READA$,PS:
C=0:FORN=1TOLEN(A$)STEP2
1201 A=ASC(MID$(A$,N,1))-48:B=ASC(MID$(A
$,N+1,1))-48:A=A+7*(A>9):B=B+7*(B>9)
1202 POKEAD,16*A+B:C=C+PEEK(AD):AD=AD+1
1203 NEXT:PRINTI;" ";:IFC<>PSTHENPRINT"E
RROR":STOP
1204 PRINT"OK":NEXT
1205 GOTO1205
1206 DATA 0D111113151115130D111113151111
100D1111131517151311100D0E101111, 541
1207 DATA 11110E0E0E0D0E10110E0D0E000B04
0A0D0D0E0E0E0D0E10110E0D11101311, 445
1208 DATA 11110A676789ED67ED890A676789ED
67676D0A676789ED3BED89676D0AA26D, 3522
1209 DATA 67676767A2A2A20AA26D67A20AA2DA
9EF7F70A0AA2A2A20AA26DA7A20A6760, 3552
1210 DATA 89676767D0040B04D0040B0B0D040B
040B040B060D040B040B050B050B060D, 637
1211 DATA 0A0D040B040B050B050B050B050B04
0B060D040B040B050B050B050B050B04, 230
1212 DATA 0B060D040B04AB45AB45AB43AB4B0A
5AB45AB45AB4B50A5AB45AB4CF9ECF9E, 3379
1213 DATA B50AB50A5AB45AB4CF9ECF9ECF9ECF
9E5AB4B50A5AB45AB4CF9ECF9ECF9ECF, 4541
1214 DATA 9E5AB4B50A5AB45A, 931
1300 POKE650,1:PRINTCHR$(142);CHR$(8)
1301 FORI=45*64+49152TO46*64+49152:POKEI
,255:NEXT
1302 FORI=46*64+49152TO47*64+49152:POKEI,
0:NEXT:POKE46*64+49152+19,3
1303 POKE46*64+49152+22,3
1304 POKE56576,PEEK(56576)AND252:POKE532
72,8:POKE648,192:POKE53270,216:POKE53282
,5
1305 POKE53283,15:POKE53269,255:POKE5327
6,255:POKE53277,255:PRINT"♥"
1306 POKE53286,0:POKE53287,5:POKE53288,1
0:POKE53285,11:POKE53284,10
1310 SYS900
```

Input de Luxe

Die INPUT-Funktion des C64 weist einige Mängel auf. So kann z.B. mit dem Cursor aus der Eingabezeile herausgefahren werden, bei Betätigung der CLR-Taste wird der Bildschirm gelöscht, Komma und Doppelpunkt können nicht eingegeben werden usw. Dies alles wird mit "Input de Luxe" umgangen. Die Länge des Eingabefeldes und dessen Position müssen bei Aufruf des Programms angegeben werden. Nach Drücken der RETURN-Taste befindet sich die Eingabe in der Variablen IN$. Innerhalb des Eingabefeldes können die Cursor-Tasten sowie HOME, CLR, DEL und INST benutzt werden. Im Großschrift-/Grafikmodus werden keine Grafikzeichen angenommen. Als Underline-Cursor wird ein Sprite verwendet. Das Programm ist voll relokatibel und kann daher an eine beliebige Stelle im Speicher gelegt werden (49152).

Funktionen: HOME: Cursor springt zum ersten Zeichen des Eingabefeldes. CLR: Eingabefeld wird gelöscht. DEL: Zeichen über Cursor wird gelöscht. INST: Einfügen einer Leerstelle. Cursortaste links und rechts. RETURN: Ende der Eingabe.

Stefan Markewitz

```
100 REM         COMMODORE 64
110 REM
120 REM  INPUT-DELUXE - RELOKATIBEL
130 REM
140 REM AUFRUF:
150 REM SYS(STARTADRESSE),X,Y,L
160 REM X=BILDSCHIRMKOORDINATE (X)
170 REM Y=BILDSCHIRMKOORDINATE (Y)
180 REM L=LAENGE DES EINGABEFELDES
190 REM
200 REM EINGABE BEFINDET SICH IN IN$
210 REM
220 REM
230 REM LAENGE DES PROGRAMMS: 481 BYTE
240 REM
250 INPUT"STARTADRESSE";SA
260 FORI=SA TO SA+481
270 READP:POKEI,P:S=S+P:NEXT
280 IFS<>61726THENPRINT"PRUEFSUMMENFEHLER":END
290 PRINT"OK.":END
300 DATA 032,253,174,032,158,183,224,040,176,095,134,020,169,000,133,250
310 DATA 138,010,010,010,105,024,141,000,208,133,249,144,007,173,016,208
320 DATA 009,001,133,250,032,253,174,032,158,183,224,025,176,059,138,010
330 DATA 010,010,105,058,141,001,208,134,113,162,000,134,114,134,041,169
340 DATA 240,133,040,032,207,179,138,024,101,020,133,020,133,003,144,001
350 DATA 200,200,200,200,200,132,021,152,024,105,212,133,004,032,253,174
360 DATA 032,158,183,134,252,232,024,144,003,076,072,178,160,063,169,000
370 DATA 153,192,003,136,016,250,169,255,141,192,003,173,021,208,009,001
380 DATA 141,021,208,169,015,141,248,007,173,134,002,141,039,208,164,252
390 DATA 169,032,234,136,208,252,132,198,132,251,196,252,240,026,169,000
400 DATA 133,251,152,010,010,010,038,251,024,101,249,141,000,208,042,041
410 DATA 001,005,251,005,250,141,016,208,132,251,032,228,255,164,251,201
420 DATA 020,240,108,201,147,240,090,201,157,240,078,201,148,240,072,201
430 DATA 019,240,187,201,029,240,056,201,013,240,117,201,032,144,217,201
440 DATA 092,144,017,232,173,024,208,041,002,208,004,124,024,144,201,104
450 DATA 201,193,144,196,201,128,144,003,041,127,044,041,063,145,020,173
460 DATA 134,002,145,003,200,196,252,144,003,136,208,142,024,144,139,200
470 DATA 196,252,144,134,136,208,131,240,057,192,000,240,001,136,024,144
480 DATA 235,164,252,136,169,032,145,020,136,016,251,169,000,240,162,152
490 DATA 072,200,196,252,176,012,192,000,240,010,177,020,136,145,020,200
500 DATA 208,239,136,169,032,145,020,104,168,024,144,192,104,136,240,249
510 DATA 240,032,132,253,164,252,136,177,020,201,032,208,016,200,136,136
520 DATA 177,020,200,145,020,196,253,208,245,169,032,145,020,164,253,024
530 DATA 144,215,173,021,208,041,254,141,021,208,164,252,136,177,020,201
540 DATA 032,208,003,136,016,247,196,252,200,132,251,174,085,165,134,012
550 DATA 202,134,013,169,073,133,069,169,206,133,070,032,231,176,133,073
560 DATA 132,074,032,163,182,165,251,032,117,180,160,002,185,097,000,145
570 DATA 073,136,016,248,165,251,240,041,200,177,020,201,160,176,022,201
580 DATA 129,176,015,201,064,176,008,201,032,144,013,044,105,064,044,105
590 DATA 127,044,105,191,044,105,127,044,105,064,145,098,200,196,251,208
600 DATA 216,096

READY.
```

PEEKs und POKEs zum Commodore 64

Von H. J. Liesert
Verlag Data-Becker
200 Seiten, 29,– DM
ISBN 3-89011-032-0

Sie haben richtig gelesen: Gegenstand dieser Besprechung ist ein Buch, das schon zwei Jahre auf dem Markt ist. Die nun vorliegende dritte überarbeitete Auflage soll uns noch einmal dazu veranlassen, kurz hineinzublicken.

Wie der Titel schon sagt, beschäftigt sich das Werk mit den PEEK- und POKE-Befehlen, die dem C-64-Benutzer aufgrund des schwachen Basic in Fleisch und Blut übergehen müssen, damit er seinen Computer voll ausnutzen kann. In leicht verständlichem Stil führt der Autor auch den Laien an die zu beherrschende Materie heran. Zunächst werden grundsätzliche Fragen über die Funktionsweise von Interpreter, Betriebssystem und weitere elementare Dinge erklärt. Dann lernt der Leser anhand vieler Beispiele mit lauffähigen Listings zum Speichern von Grafiken, der Programmierung von Sprites oder der Erstellung eines eigenen Zeichensatzes den Umgang mit Speicherstellen und ihre Anwendung kennen.

Wie nützlich und gut das Buch ist, hat es in der Vergangenheit durch hohe Verkaufszahlen bewiesen. Einige kleine Fehler, die sich eingeschlichen hatten, sollen in der vorliegenden Auflage vollständig beseitigt sein. Das Buch wurde um einen Speicherbelegungsplan im Anhang erweitert, der alle wichtigen Speicherstellen kurz erklärt. Dem eigenen Experimentieren mit PEEKs und POKEs steht also nichts mehr im Wege.

Thomas Tai

Pascal mit dem C 64

Von Florian Matthes
Markt & Technik Verlag
220 Seiten, 52,– DM
ISBN 3-89090-222-7

Ein günstiges Angebot von Markt & Technik ist "Pascal mit dem C 64". Hierbei handelt es sich um einen Pascal-Compiler/Editor, der nicht nur den Ansprüchen eines Anfängers genügen sollte, und um ein rund zweihundertseitiges Buch, das außer einer ausführlichen Einleitung in diese Programmiersprache auch noch Tips und Tricks für Fortgeschrittene bietet.

Der Pascal-Kurs ist – wie bei dieser Sprache mit ihren strikten Formen nicht anders zu erwarten – anfangs recht trocken. Man wird in die grundsätzliche Systemstruktur mit Editor und Compiler eingeführt, lernt Syntaxdiagramme kennen und kämpft sich mit ihrer Hilfe von einfachen Operationen (Ein- und Ausgabe, Auswerten von Ausdrücken) über Fallunterscheidungen bis zu den vielfältigen Schleifentypen durch. Besonders ausführlich werden dabei die für Pascal typischen komplexen Strukturen wie Prozeduren und Funktionen, hauptsächlich aber Rekursionen erklärt. Sachlich und nicht besonders originell, dafür aber gut verständlich, erläutert die Einführung selbst Sortieralgorithmen und Datenstrukturen wie die bekannte Baumstruktur.

Im darauffolgenden Teil (Tips & Tricks) erlernt man den Umgang mit Files zum Erstellen eines Directory-Programms und wird mit Systemadressen sowie der Einbindung von Maschinensprache in Pascal-Programme vertraut gemacht. Zusätzlich ist eine ausführliche Dokumentation zum beigelegten Compiler/Editor enthalten. Im Anhang findet man Listen von Syntaxdiagrammen, Fehlernummern, Laufzeitfehlern und Operatoren sowie einige Literaturhinweise. Eine Kurzzusammenfassung für Programmierer, die eine bestimmte Information suchen, fehlt leider.

Auf der Programmdiskette, die zusätzlich noch sechs beachtliche Demoprogramme enthält, findet sich das aus Editor, Compiler und Laufzeitsystem bestehende Pascal-System.

Der Editor erinnert stark an den gewöhnlichen Full-Screen-Editor des C 64, hat jedoch einige Vorteile aufzuweisen. So kann zum Beispiel die maximale Länge einer Programmzeile zwischen 0 und 80 frei gewählt werden; mit den Funktionstasten bewegt man dann das Bildschirmfenster wie über ein Blatt Papier. Außer der Möglichkeit, Tabulatoren festzulegen, bietet er auch noch sogenannte Primary und Line Commands an, die dazu dienen, den Editor zu verlassen, den Source-Text ganz oder teilweise auf Diskette zu schreiben bzw. zu lesen und Zeilen einzufügen, zu löschen, zu kopieren oder zu verschieben. Es gibt auch zahlreiche Blockbefehle, mit denen einzelne Textblöcke verarbeitet werden können, sowie die nützlichen Befehle Find und Change, die längeres Durchsuchen des Source-Textes ersparen.

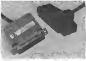
Der Compiler übersetzt den Quelltext im Speicher, was eine enorme Zeitersparnis auf Kosten des Speicherplatzes bedeutet, und hinterläßt das Object-Programm im Basic-Speicher, wo es beliebig gestartet, gesichert und auch ohne das System geladen werden kann. Er bietet außer der Übersetzung noch die Kontrolle auf Syntaxfehler und die Möglichkeit, die Position von Runtime-Errors im Source-Text festzustellen, übersetzt mit wenigen Ausnahmen den gebräuchlichen Standard (bedeutendste Ausnahme ist eine einfache Potenzierungsfunktion) und reagiert auf aktive Kommentare (bei Assemblern würde man sie Pseudo-Opcodes nennen), die zum Beispiel das Mitübersetzen von Quellprogrammen auf der Diskette erlauben. Dies ist bei nur 8 KByte Sourcecode, die in den Speicher passen, wirklich kein Luxus. Da das gut verständliche Lehrbuch und der durchaus brauchbare Compiler/Editor zusammen zu einem annehmbaren Preis angeboten werden, ist das System für Anfänger uneingeschränkt empfehlenswert.

Axel Kleck

V24-Adapter für C 64/C 128

C 64/C 128-Computer haben eine eingebaute serielle Schnittstelle, die mit TTL-Pegel auf den User-Port geführt ist. Mit Hilfe des Interface Typ 89002 werden die TTL-Pegel in normgerechte V24-Signale gewandelt (ca. ±9V). Die notwendige Elektronik einschließlich Spannungswandler ist sehr kompakt im Steckergehäuse untergebracht. Eine Stromversorgung ist nicht erforderlich. Preis ca. 98,– DM.

Weitere Informationen:
Wiesemann Mikrocomputertechnik
Winchenbachstraße 3-5
5600 Wuppertal 2



Einfaches, aber solides Innenleben

Amiga-Spieleübersicht

Wer die Leistungsdaten des Amiga betrachtet, denkt sofort: das ist der ideale Spielcomputer. Obwohl Commodore es gar nicht so gerne sieht, daß ihr Flaggschiff in diesen Ruf kommt, tun die Softwarehäuser doch alles, um den Amiga-Besitzer zum Spielen zu animieren. Dabei handelt es sich zum Teil um Adaptionen älter 64er Programme, aber auch um völlig neue Spiele mit neuen Ideen.

Die Programmierer der ersten Amiga-Spiele hatten sich nicht gerade mit Ruhm bekleckert, als sie C-64-Spiele umschrieben, ohne auf die besonderen Fähigkeiten dieses "Mediencomputers" einzugehen. Dies bessert sich aber zusehends. Immer mehr Spiele werden auf den Markt geworfen und gerade zur Weihnachtszeit arbeiten die Kopiermaschinen auf Hochtouren. Deshalb ist es recht nützlich, sich einen kleinen Überblick über das aktuelle Angebot zu verschaffen. Die nun folgende Zusammenstellung von Spielen für den Amiga erhebt keinen Anspruch auf Vollständigkeit. Einige Programme wurden aufgrund ihrer schlechten Umsetzung bzw. Qualität bewußt weggelassen, andere waren zur Zeit der Entstehung dieses Artikels noch nicht auf dem Markt. Das Fehlen eines Programms in dieser Liste bedeutet also nicht unbedingt, daß es schlecht ist!

Brataccas

Das erste Spiel, das ausschließlich für die neue 68000er Generation geschrieben wurde, basiert auf einer völlig neuen Spielidee. Auf dem Bildschirm läuft ein richtiger Zeichentrickfilm ab, in den Sie eingreifen können. Es ist egal, was Sie tun und ob Sie etwas tun, "Brataccas" läuft unbeirrt weiter. Der Spieler schlüpft in die Rolle des Soldaten Kyne, der nach einer Reihe von unglücklichen Ereignissen in das Kreuzfeuer zwischen Unterwelt und Regierung gerät. Alle Bemühungen, seine Unschuld zu beweisen, führen ihn auf den Planeten Brataccas.

Die Steuerung wird mit der Maus abgewickelt und ist recht gewöhnungsbedürftig. Im gesamten Spiel ist nichts genau festgelegt, es handelt sich um "Symbol Action", d.h. das Programm interpretiert Ihre Mausbewegung in der aktuellen Situation z.B. als Bewegung, Handlung oder Kommunikationsversuch.

"Brataccas" besticht durch seine interessante und komplexe Handlung. Grafik und Sound bleiben allerdings weit hinter den Möglichkeiten des Amiga zurück. Es ist aber trotzdem ein empfehlenswertes Spiel für die Freunde von Action Adventures.

System	Amiga 512 K
Preis	ca. 119,- DM

Little Computer People Project

Die Story von dem kleinen Männchen im Computer ist ja nicht neu, und man hoffte deshalb auf eine spektakuläre Umsetzung. Doch sie blieb leider aus: man erhält biedere Hausmannskost. Das Männchen in Blockgrafik bewegt sich ruckartig, die Geräusche sind blechern computerhaft, eben ganz so, wie sie beim Amiga nicht sein müßten. Der Reiz dieser hervorragenden Spielidee bleibt natürlich erhalten, doch in Anbetracht dessen, daß das Programm nur um ein Kartenspiel im Vergleich zur C-64-Version erweitert wurde, kommen doch arge Zweifel, ob man sich genug Mühe bei der Programmierung gegeben hat.

"Little Computer People" ist aber sicherlich ein Erlebnis, wenn man solch ein Programm noch nicht kennt. Der kleine Kerl auf dem Bildschirm geht einem ganz normalen Leben nach: Zähne putzen, fernsehen, Computer programmieren, essen, trinken, abwaschen, Hund füttern, tanzen, Klavier spielen. Auf Wunsch tritt er sogar mit uns in Verbindung. Wer das Spiel bereits kennt, sollte sein Geld besser anders anlegen.

System	Amiga 512 K
Preis	ca. 109,- DM

One On One

Eine der ersten ECA-Umsetzungen für den Amiga ist dieses Basketballspiel vom C64. Leider ist die Grafik von sehr schlechter Qualität. Der Sound allerdings entschädigt für so manches: Zuschauergemurmel, Ballgeräusche, Turnschuhquietschen, Schiedsrichterpfiffe und aufbrausender Jubel bei jedem Korb klingen da aus den Lautsprechern und versetzen den Spieler wirklich in eine Basketballhalle. Der hier verwendete digitalisierte Sound läßt erahnen, was in Zukunft noch auf dem Markt erscheinen wird. Das Ganze ist allerdings keine richtige Simulation des Basketballspiels, denn es spielen nur zwei Gegner auf einen Korb. Die Steuerung mit der Maus ist kaum möglich. Joysticks sollten auf jeden Fall genommen werden, wenn zwei Spieler gegeneinander antreten.

Jeder, der über die schlechte Grafik hinwegsehen kann, sollte sich "One On One" ruhigen Gewissens zulegen.

System	Amiga 256 K
Hersteller	ECA
Preis	ca. 90,– DM

Marble Madness

"Was lange währt, wird endlich gut", lautet ein altes Sprichwort, das in diesem Spiel wieder einmal seine eindrucksvolle Bestätigung fand. Monatelang war "Marble Madness" ein Zauberwort für Spiele-Freaks, das ihre Augen zum Leuchten brachte. Jetzt endlich ist es da, und wer es gesehen hat, weiß, daß alle Erwartungen weit übertroffen wurden.

Fantastische 3-D-Grafik in atemberaubend schnellem Scrolling, witzige Effekte und ein Supersound sind für das Spiel bezeichnend. Die Spielidee ist überraschend simpel: Eine Kugel muß durch 8 Level an Abgründen, Säuretümpeln, Murmelfressern und Staubsaugern vorbei heil ins Ziel gebracht werden. Wer sich erst einmal vor den Computer gesetzt hat, wird von "Marble Madness" nicht so schnell wieder loskommen. Unzählige sehenswerte Effekte und eine hohe Motivation treiben den Spiele-Freak zu Höchstleistungen an. Es handelt sich wohl um das beste Amiga-Spiel, das bisher das Licht der Welt erblickt hat.

System	Amiga 512 K
Preis	ca. 90,– DM

The Seven Cities of Gold

Noch eine der C-64-Umsetzungen auf den Amiga von ECA. Dieses Spiel ist allerdings nur aufgrund der relativ interessanten Handlung empfehlenswert. Sie bekommen ein Schiff und eine bestimmte Geldmenge, sollen neue Länder entdecken und natürlich Reichtümer zurückbringen. Das Zusammenstellen der Ausrüstung will gut überlegt sein, bevor es dann auf die große Reise durch die Weltmeere geht.

Weder Grafik noch Sound entsprechen den Möglichkeiten des Amiga. Selbst erfahrene Spiele-Freaks werden kaum Unterschiede zwischen der Version für den C64 und der für den Amiga feststellen können.

System	Amiga
Hersteller	ECA
Preis	ca. 109,– DM

Rogue

Ein typisches Labyrinthspiel, das auch gleichzeitig für den Atari ST herauskam. Die beiden Versionen unterscheiden sich trotz der unterschiedlichen Hardware kaum voneinander. Der Spieler steuert einen mutigen Kämpfer durch 26 Levels, in denen es zunächst einmal nur um das Überleben gegen Monster und sonstige Kreaturen der Hölle geht. Wer allerdings tiefer in das Programm einsteigt, wird sich auch bald an die eigentliche Aufgabe heranwagen: Die Suche nach dem Amulett von Yendor.

"Es ist sicherlich das beste Labyrinth-Spiel, das es derzeit für den Amiga gibt. Grafik und Sound sind hervorragend, die Programmierer gaben sich offenbar Mühe.

System	Amiga 512 K
Preis	ca. 89,– DM

Archon

Der Kampf des Guten gegen das Böse ist so alt wie die Menschheit. Zahlreiche Legenden hatten Einfluß auf Rollenspiele und Adventures für den Computer. Archon ist nun das erste Strategiespiel, das sich mit diesem Komplex befaßt. Die Umsetzung vom C64 ist recht gut gelungen, vor allem der hervorragende Sound des Spieles erwartet. Durch geschicktes Einsetzen seiner Figuren werden wichtige Kraftquellen besetzt und Kämpfe zu eigenen Gunsten entschieden.

Insgesamt erscheint "Archon" wie die Variante eines abgemagerten Schachspiels, und tatsächlich hat man vor allem zu zweit viel Spaß. Es gehört zu den wenigen empfehlenswerten Spielen für den Amiga.

System	Amiga 512 K
Preis	ca. 109,– DM

Hacker II

Der Nachfolger des legendären "Hacker", das die 8-Bit-Spielwelt noch heute beschäftigt, ist wieder ein hochwertiges Produkt von Steve Cartwright.

Im Auftrag des CIA steuert der berühmte Hacker einen kleinen Roboter in einem Hochsicherheitstrakt von Sibirien, in dem geheime Dokumente über die Vernichtung der USA lagern.

Abgesehen von der sehr politischen Handlung, mit der sich wieder ein hervorragendes Feindbild gegen die "bösen Russen" aufbauen läßt, ist "Hacker II" ein würdiger Nachfolger des ehemaligen Super-Hits. Man muß aber zur Lösung schon einige Stunden geistige Arbeit investieren. Ein Vergleich mit dem Schwierigkeitsgrad von "Hacker I" ist auf keinen Fall möglich. Für Profis mit gehobenen Ansprüchen ist "Hacker II" das ideale Spiel, mit dem der Amiga seinen Ruf als Spielcomputer erstmals erfolgreich verteidigen kann.

System	Amiga 512 K

Nun noch zu zwei Spielen, die kürzlich angekündigt wurden.

Leader Board

Das altbekannte Golfspiel vom C64 wurde hervorragend auf den Amiga übertragen. Sandbunker, abwechslungsreiche Grafik, kurz alles, was der ersten Version fehlte, hat nun dieses Programm.

Silent Service

Als U-Boot-Kommandant im Zweiten Weltkrieg geht es mitten hinein in die geheimnisvolle Welt der Unterwasser-Abenteuer. Solche "kriegerischen Spielereien" sind allerdings nicht jedermanns Sache, und daher sollte man dieses Programm einmal kurz anspielen, bevor man es erwirbt.

Der Trend auf dem Softwaremarkt für den Amiga geht eindeutig zum Spiel. Jedes annähernd erfolgversprechende Programm wird für ihn umgesetzt. Zu erwarten sind noch der "Game Maker" von Activision, "Deep Space", ein Elite-Clone von Psygnosis, und die Sportspiele "Arena" und "Winter Games".

Eines sollten Sie also bei allem ernsthaften Arbeiten mit dem Amiga nicht vergessen: Ihr Computer kann auch spielen, und wenn das Niveau weiterhin so rapide ansteigt, stehen wir bald vor einer Flut von hervorragenden Programmen, die man nicht nur spielt, sondern wirklich anwendet (z.B. zur Entspannung nach anstrengender Arbeit).

Thomas Tai

Lattice auf Amiga-Pfaden

Das renommierte amerikanische Softwarehaus Lattice, das Software-Entwicklern mit unzähligen Utilities und Compilern unter die Arme gegriffen hat, bringt nun auch eine Produktpalette für den Amiga heraus. An dieser Stelle soll zunächst nur ein kleiner Überblick über die Programme gegeben werden. Ein ausführlicher Test und eine Beurteilung folgen später.

Für den C-Programmierer dürfte natürlich der C-Compiler interessant sein, der allerdings schon etwas länger auf dem Markt ist und nur der Vollständigkeit halber erwähnt sei. Schon beliebter wird man bei der "Mac Library", die 60 Funktionen zur Einbindung in C-Programme enthält. Mit ihr kann man die Grafikfähigkeiten des Amiga besser nutzen und spart sich das Zusammenstellen eigener Funktionen aus mühsam herausgesuchten Adressen. So ganz nebenbei sind die Funktionen auch Macintosh kompatibel. Wer also C-Programme vom Macintosh benutzt, kann diese ohne Probleme auf den Amiga übertragen.

Die "dBase III Library" soll eine Verbindung zum Standardprogramm der MS-DOS Welt "dBase III" schaffen. Die Funktionen werden wieder von C benutzt und ermöglichen den Aufbau von Dateien, die "dBase III" lesen und auswerten kann, sowie das Erweitern von bereits existierenden "dBase III" Dateien. Der Umstieg von MS-DOS auf den Amiga wird somit noch leichter, da die Datenbestände weiterhin genutzt werden können.

Die folgenden Utilities sind wieder für alle Programmierer interessant, ganz gleich welche Sprache sie bevorzugen. Mit dem "Lattice Screen Editor" können Programme komfortabel und einfach wie mit einer Textverarbeitung editiert werden. Herausragendes Leistungsmerkmal ist hier die Multiwindow-Funktion, mit der mehrere Dateien gleichzeitig bearbeitet werden können. Zwischen ihnen ist auch ein Austausch von Zeichenfolgen möglich. Ebenfalls für die effektivere Arbeit mit Texten sind die "Lattice Text Utilities" gedacht. Mit dieser Sammlung von acht Programmen können Worte einer Datei gezählt, Zeichenfolgen gesucht und ersetzt, Dateien verglichen und Unterschiede aufgelistet sowie Batch-Dateien erzeugt werden.

Soviel für diesmal von den Lattice-Produkten. Dem interessierten User sei noch gesagt, daß die Firma Pfotenhauer Microcomputer Anwendungen den Generalvertrieb der Produkte übernommen hat und deutsche Versionen in Arbeit sind.

Thomas Tu

Anzeigenschluß

für Nr. 2/3-87 Computer Kontakt ist der 2. Januar 1987.

Kuma-Seka, ein Macro-Assembler für den Amiga

Assembler Programme für den Amiga sind vorerst noch rar. Umso erstaunlicher ist es, daß sich im kleinen Assembler ein Programm für den Commodore-Knüller befindet, das kaum Wünsche offenläßt. "Seka" heißt die angenehme Überraschung aus England, die von der Computer-Firma Kuma auf den Markt gebracht wurde und mit Sicherheit viele Freunde finden wird.

"Seka" ist recht einfach in der Handhabung. Der Assembler wird aus dem CLI heraus mit Namen aufgerufen, es genügt, "SEKA" einzugeben. Nach kurzer Ladezeit verlangt das Programm nach der Workspace-Angabe. Danach wird der User gefragt, wieviel Arbeitsspeicher dem Assembler zugeteilt werden soll.

Nun zu den Besonderheiten des Assemblers. "Seka" bietet einen Zeilen- sowie einen Screen-Editor, zwischen denen mit der ESC-Taste hin- und hergeschaltet werden kann. Der Zeilen-Editor verfügt über alle üblichen Funktionen, wie INSERT, EDIT, LOCATE usw., beim Screen-Editor sind CUT, COPY und PASTE selbstverständlich. "Seka" ist in der Lage, absoluten oder voll relokativen Code zu erzeugen, und verfügt über 30 Pseudo-Operatoren.

Der Befehl ORG veranlaßt den Assembler, absoluten Code zu erzeugen, der durch LOAD an die Adresse geladen wird. Sollte jedoch CODE eingegeben werden, so wird relokativer (soll im Speicher verschiebbarer) Code erzeugt. Durch EVEN wird die nachfolgende Anweisung auf eine gerade Adresse gesetzt, da der MC 68000 nur auf solche zurückgreifen kann. Mit der Macro-Anweisung definiert man neue Befehle. "Seka" verfügt auch über Befehle, die den Drucker steuern. Bedingungen für die Assemblierung können durch IF, ELSE und ENDIF geschaffen werden. BLK hingegen definiert einen Block im Speicher, der mit einem Byte, Wort oder Langwort gefüllt wird.

Das wichtigste Kommando des Assemblers ist zweifellos A, das für assemble steht. Nachdem das Kommando eingegeben ist, verlangt "Seka" nach einigen Optionen. Werden diese nur mit RETURN beantwortet, so gibt er nur die fehlerhaften Zeilen aus. Die Option V steht für Video und bringt das Listing auf den Bildschirm. E und P bewirken seine Ausgabe auf einem Drucker. Sehr interessant ist auch die Option O. Alle Macros ohne 0 oder 1 werden im Source optimiert (z. B. wird an jeden Sprungbefehl, soweit dies möglich ist, ein S angefügt).

Bemerkenswert ist auch der Befehl WO (Writeobject). Mit ihm wird das assemblierte File, das relokativ sein muß, als ein vom CLI aufrufbares Programm abgelegt. Dieses kann auch von der Workbench aus aufgerufen werden, sofern es mit einem Icon ausgestattet wird, oder als Basic-Routine, wenn es als bmap-Datei im libs-Directory abgelegt wurde.

Ein auf der "Seka"-Disk mitgeliefertes Programm "amiga dos.i" wird zu einem Source-File geladen. In diesem Programm steht eine Reihe von DOS-Befehlen wie z. B. OPEN, CLOSE, READ und WRITE, die mit JSR aufgerufen werden. So ist es möglich, mit einer MC-Routine eine andere zu laden, zu löschen oder zu erzeugen – und dies alles mit ein paar DOS-Aufrufen.

"Seka" verfügt auch über einen eingebauten Disassembler. Für den Single-Stepping und Modifizieren aller Register kein Problem darstellen. Alle Standardbefehle, z. B. für das Kopieren von Speicherblöcken, Sprüngen zu einer Subroutine oder Starten ganzer Programme, sind vorhanden.

Das Handbuch des Assemblers umfaßt 35 Seiten. Die Erläuterungen sind so gut verständlichem Englisch gehalten, so daß man sich ohne weiteres zurechtfinden kann.

Bert Zeittner

SEKA
AMIGA
Kuma

Einfach in der Handhabung durch den Full-Screen Editor, auch ein Disassembler ist dabei

ATARI

[illegible]

Peeks & Pokes zu Atari 400 XL/800 XL

[illegible]

Bestellnummer BB 1 DM 38,-

D. [illegible]

Atari mit Atari-Logo

[illegible]

Bestellnummer V 8 DM 38,-

A. Hettinger · A. [illegible]

Atari mit Atari Basic

[illegible]

Bestellnummer V 9 DM 38,-

A. + J. [illegible]

Was der Atari alles kann
Band 1

[illegible]

Bestellnummer V 4 DM 38,-

A. + J. [illegible]

Was der Atari alles kann
Band 2

[illegible]

Bestellnummer V 6 DM 38,-

A. Hettinger · V. Krauß

Die Atari-Hitparade

[illegible]

Bestellnummer V 8 DM 38,-

[illegible]

Lerne Basic auf dem Atari

[illegible]

Bestellnummer MT 1 DM 38,-

L. H. Schreiber

Das Atari Programmier handbuch

[illegible]

Bestellnummer MT 3 DM 58,-

M. [illegible]

Das Schulbuch zu Atari 800 XL/600 XL

[illegible]

Bestellnummer DW 8 DM 48,-

Norbert Szczepanowski

Atari 130 XE, 800 XL, 600 XL

[illegible]

Bestellnummer DW 10 DM 39,-

Stanley R. Trost

Atari-Programm-Sammlung

[illegible]

Bestellnummer SY 11 DM 34,-

Poschke/Wiethoff

Das Atari Profibuch

[illegible]

Bestellnummer SY 22 DM 48,-

Julian Reschke

Atari Basic Handbuch

[illegible]

Bestellnummer SY 10 DM 38,-

Rainer Schmidt

Spielen, lernen und arbeiten mit dem Atari

[illegible]

Bestellnummer SY 14 DM 38,-

Tom Rowley

Sprühende Ideen mit Atari Grafik

[illegible]

Bestellnummer TW 6 DM 48,-

[illegible]

Atari 600 XL/800 XL Intern

[illegible]

Bestellnummer DW 14 DM 48,-

[illegible]

Das Basic Trainingsbuch zu Atari 600 XL/800 XL

[illegible]

Bestellnummer DW 17 DM 39,-

Don Inman/Kurt Inman

Der Atari Assembler

[illegible]

Bestellnummer DB 10 DM 39,-

BUCHVERSAND

Schneider

Strategiespiele, und wie man sie auf dem Atari 600 XL/800 XL programmiert

[illegible]

Bestellnummer OS 10 DM [illegible]

Poole McNiff Cook

Mein Atari-Computer

[illegible] Seiten

Ein Handbuch, das für jeden Atari-Besitzer [illegible]

Bestellnummer TW [illegible] DM [illegible]

Steiner

GEM für den Atari 520 ST

344 Seiten

Dieses Werk ist eine Einführung in [illegible]

Bestellnummer MT [illegible] DM [illegible]

Jürgen [illegible]

Wordstar für den Atari ST

[illegible] Seiten

Dieses Buch ist [illegible]

Bestellnummer MT [illegible] DM [illegible]

[illegible] Apple

Lernen mit Spaß – 30 Lernspiele für den Atari

[illegible]

Bestellnummer [illegible] DM [illegible]

Alfred Görgens

Utilities in Basic für Atari-Computer

[illegible]

In diesem Buch finden [illegible]

Bestellnummer V 28 DM [illegible]

[illegible]

Das große Spielebuch für Atari Band 1

[illegible]

Bestellnummer [illegible] DM [illegible]

[illegible]

Das große Spielebuch für Atari Band 2

[illegible]

Dieses Buch enthält [illegible]

Bestellnummer [illegible] DM [illegible]

[illegible]

Adventures, und wie man sie auf dem Atari 600 XL/800 XL programmiert

[illegible]

Bestellnummer [illegible] 37 DM [illegible]

[illegible]

Atari als Texter

[illegible]

Bestellnummer RV 20 DM 34,–

Rugg Feldman Barry

20 Basic Programme für den Atari

[illegible]

Das Buch enthält [illegible]

Bestellnummer [illegible] 39 DM 34,–

James Dale Eubank

Das Atari Spielebuch für 600 XL/800 XL

[illegible]

Bestellnummer V [illegible] DM [illegible]

Buch-Bestellschein

Bitte liefern Sie mir folgende Bücher

Anzahl	Bestell-Nr.	Einzel-Preis

Ich wünsche folgende Bezahlung

☐ Nachnahme (+ 5,70 DM Porto + Versandkosten)

☐ Vorauskasse (keine Versandkosten)

Bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe [illegible] überweisen

Name des Bestellers

Anschrift

PLZ/Ort

Telefon

Datum/Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden: Verlag Rätz-Eberle Postfach 1640 7518 Bretten

CK-PROGR

SINCLAIR

Paint (nicht veröffentlicht) Pyramide (7/84) Superfilm (7/84) Dancer (9/84) Blutzuckerdiagramm (10/84) Große Buchstaben (10/84) Farben im Spectrum (10/84) Prodomo (11/84) Fonts (12/84) Uhrzeit (12/84) 3-D Schrift (12/84) Neuer Zeichensatz (12/84) Krümelmonster (1/85) Font I, S (1/85)
Best.-Nr. S 1

Fusee (4/85) Spiffex mit Demo (3/85) Darts (5/85) Uhr (7/85) Roulette
Best.-Nr. S 10

Catalog (2/85) Softsave (2/85) Hiresline (2/85) Computer figures (2/85) Ku Bemd (2/85) Sterngrafik (3/85) Mimic latin (3/85) Ieran (4/85)
Best.-Nr. S 11

Schattens (5/85) Spectrum-Infosystem (6/85) Cowboy (4/85) Player's Dream (6/85) Tape (6/85) Soundsampler (3/85) Hovercraft (7/85)
Best.-Nr. S 20

ATARI

Lunar Lander (12/84) Cerberus (7/84) Turbo Worm (1/85) Mundorgel (3/85) Bewegte Grafik (3/85) Bigger (3/85) 15 und 3 (4/85) Bondesignalmusiken (3/85) 3-D Laby (10/84) Zeichensatzeditor (2/85) Mini-Malstudio (9/85) Bolly Dolly (11/84) Musik-Editor (4/85), Hi nur bei Kassettenversion Programme sind nur mit Erweiterung lauffähig
Best.-Nr. A 10 Diskette Best.-Nr. A 10a Kassette

Sound-Demo (5/85) Sound-Rahmen (nicht veröffentlicht) The Run and Jump Construction Set (4/85) Bank Punk (7/85) nur mit Funktions-Plotter (6/85) Softcube (7/85) Jewel Editor (5/85) Zeichen-Assembler (7/85) Joystick-Controller (9/85) Horizontales-Scrolling (5/85) Converter (DOS 1 in DOS 1) (9/85) Nur auf Diskette
Best.-Nr. A 11 Diskette Best.-Nr. A 11a Kassette

TI 99/4A

Burg in Time (12/84) Cowboy (7/84) Desert Night (9/84) Fassadenkleftern (11/84) Hungman (nicht veröffentlicht) Miner-Pal (4/85) Novo-Montage (1/85) Fallschule Jumper (5/84) Permanente Kleinbuchstaben (10/84) Pokalistengenerator (12/84) Screen Utilities (11/84)
Burglar Time (12/84) Cowboy (7/84) Desert Night (9/84) Fassadenkleftern (11/84) Hungman (nicht veröffentlicht) Miner-Pal (4/85) Novo-Montage (1/85) Fallschule Jumper (6/84) Permanente Kleinbuchstaben (10/84) Pokalistengenerator (12/84) Screen Utilities (11/84)
Best.-Nr. TI 1 Diskette TI 1m Kassette

Aloha Look (2/85) Cube (3/85) Epsstel (4/85) Jungler (4/85) Macropede (4/85) Merge-Filer (3/85) Motor ON (2/85) Penyrun (2/85) Prog neid (3/85) Rotation (2/85) Vokanel (2/85), Achtung: Macropede Merge-Filer und Prog neid nur auf Diskette
Best.-Nr. TI 10 Diskette Best.-Nr. TI 10a Kassette

Basisette pour Aneline (nicht veröffentlicht) Kugelmuster (6/85) Super Disk-Katalog (4 und 7/85) Fluch des Pharao (4/86) Plot (6/85) Säulendiagramm (6/85) Calculator (7/85) Texter (6/85) Würfel-Duell (7/85) Nur auf Diskette
Best.-Nr. TI 11 Diskette, Best.-Nr. TI 11a Kassette

Gold Jump (10/85) Cavern (9/85) Crazy Man (11/85) Dreo Quest (10/85) Eliert-Sin (11/85) High Sea, Grafik mit Demo (11/85) Soundeditor (10/85) Sprite-Utilities (11/85) Seltor II (9/85) Nur auf Diskette
Best.-Nr. TI 12 Diskette, Best.-Nr. 12a Kassette

COMMODORE

Duell (7/84) Mäuserennen (9/84) Speicherplatzanzeige (11/84) Basic-Erweiterung (10/84) Through the wall (11/84) Maze-Bot (11/84) Prüfsummengenerator und Indikator (11/84) Grafik-Erweiterung (12/84) Beat die (12/84) Photons (1/85) Nürburgring (1/85)
Best.-Nr. C 1 Diskette DM 20,– Best.-Nr. C 2 Kassette DM 18,–

Handmaßmanager (3/85) Defender (2/85) Ghosts (4/85) Pagna (4/85) Delete (4/85) Merge (2/85) Find (2/85) Screen-Designer (3/85) Set Des Orig Color (2/85) Drehwerk (2/85) Kofferuntersucher (3/85) Auto-Starter 1.0 (6/85) Code 64 (5/85) File-Load (5/85) File-Save (5/85)
Best.-Nr. C 10 Diskette DM 20,– Best.-Nr. C 10a Kassette DM 16,–

Die Grafikerweiterung Set und 3 sowie die Basic-Erweiterung EXS V 1.0 sind nur auf Diskette. Kassette inklusive Anleitung bezogen werden. (Siehe auch CK-Hefte 11/84, 12/84 und 1/85)
Best.-Nr. DM88 Diskette DM 13,– Best.-Nr. K886 Kassette DM 10,–

FREIE AUSWAHL

3er Pack	**5er Pack**
3 Kassetten nach Ihrer Wahl nur **40,–**	5 Kassetten nach Ihrer Wahl nur **60,–**
3 Disketten nach Ihrer Wahl nur **50,–**	5 Disketten nach Ihrer Wahl nur **75,–**

Je
D
J

Je
D
J

Je
D
J

Ne
Pr
C
D
K

M M S E R V I C E

assette
16,–

Paint (9/85) Window (9/85) 3-D Plotter (10/85), Superfast (10/85) Centipede (11/85) Nitro Jue (11/85)
Best.-Nr. 3 38

Höllensturz (1/86) The Seventh (1/86) Snake-Smasher (2/86) Axel (3/86) Uni LLIST (3/86) Programmzeilen löschen (3/86) Spectrum-Quick copy (2/86) On Error Goto (3/86)
Best.-Nr. 3 40

Ass. Tips Uhr (7/86) GLP-Druckerroutine (7/86), Datenverschlüsselung (7/86) Pro Dos (7/86) Apfelmännchen (7/86) Tele (7/86) Diskmaster (5/86) Spectrape (5/86)
Best.-Nr. 3 46

Kassette
16,–

neu

Assemblertips Teil 1 (9/86) Aufsteckkiller (9/86) Vorschleinere (9/86) Headerchange (9/86) Assemblertips Teil 10 (11/86) Jahrmarkt (11/86) Schreibschrift (11/86) Discopen Diskettenwert (11/86) Apfelmännchengrafik (11/86) Mini Wolkkomme (11/86) Hardcopy Epson 8 (11/86) Grafikeditor (1/87) Quasicopy (1/87) Überschrift (1/87) Window (1/87)
Best.-Nr. 3 56

Diskette
20,–

Kassette
16,–

Dt. Designer 64K (10/86) Joypaint 64K (10/86), Musicsatur 64K (11/86) Charakterdekleur 64K (1/85) Ur protector V1.0 16K (1/86) Key Maker 16K (1/86)
Best.-Nr. A 12 Diskette, Best.-Nr. A 12a Kassette

Cherry Henry (3/81) Mission X auf dem Atari (1/86) Basic-Erweiterung (3/86) Mini-Bildeni (10/86) Zeichen-Zauberer (3/86) Sound-Demo (3/86) Nur auf Diskette
Best.-Nr. A 13 Diskette, Best.-Nr. A 13a Kassette

Kassette Kid (1/86) Fyt-Dos (7/86) Text im Grafikdreieler (7/86) Silhouette (7/86) Kung Fu (9/86) Disk-Menü (9/86) Tron (9/86) Nur auf Diskette
Best.-Nr. A 14 Diskette, Best.-Nr. A 14a Kassette

neu

Der hungrige Golf (11/86) Atari-Puzzler (11/86) Karteiverwaltung (11/86) Disc-Collector (11/86) Mini Disk Programm (11/86) Mic nMon (nur für Kassettenbetreiben) (11/86) Wermitel (1/87) Cnc 800 (1/87) Diskeditor (1/87) Speed Tape (1/87) Filecopy (1/87) Zeichensatzlinear (1/87) Hardcopy-GP 500 AT (1/87) Nur auf Diskette
Best.-Nr. A 15 Diskette, Best.-Nr. A 15a Kassette

Diskette
20,–

Kassette
16,–

Bierbörse (3/86) Buchem (3/86) Copy.T (1/86) Hardcopy für Seikosha GP 100 (1/86) Lander (3/86), Moondiver (5/86) Polargrafik (3/86) TI-Point (3/86) 40 Zeichen-Mode (3/86) Turtle Jumper (1/86) Nur auf Diskette
Best.-Nr. TI 13 Diskette, Best.-Nr. TI 13a Kassette

TI-Assemblerdiskette 1 Für die TI-Assemblerfreaks haben wir auf vielfachen Wunsch eine Diskette mit den Quellcodes der ehemaligen Assemblerprogramme Macropede (4/86) und Super Disk Catalog (1 und 2/86) zusammengestellt. Zum Ändern und Lernen
Best.-Nr. TI 14 Diskette

TI-Assemblerdiskette 2 Diese Diskette enthält die Quellcode-Dateien für die Assemblerprogramme Adressdatei (4/86) und Disk Utilities (9/86)
Best.-Nr. TI 15 Diskette

Willy der Wurm (7/86) Basicshow GP 50 Hardcopy (7/86) Snoopy (7/86) Adressdatei (5/86) Soundeditor plus (9/86) Screen Editor (7/86) Assembler-Grafik-Generator (7/86) The Plague (9/86) Disk Utilities (9/86) Nur auf Diskette
Best.-Nr. TI 16 Diskette, Best.-Nr. TI 16a Kassette

neu

Monopoly (7/86) Maschinenkatalog (11/86) Buchstaben-Mix (11/86) Der Maler (11/86) Textverschlüsselung (11/86) Combora (1/87) Lord Bell Going (1/87) U-Boot Kampf (1/87) SROC 2 (1/87) Hexmonitor in C (1/87) Nur auf Diskette
Best.-Nr. TI 17 Diskette, Best.-Nr. TI 17a Kassette

Sonder-
für alle
nodare
tten und
etten

Baloon (7/85) Rescue in the Stone-Age (1/85) Solitaire (7/86) Radio Editor (9/86), Data Loader (8/86) Rahmenroutine (7/86)
Best.-Nr. C 11 Diskette DM 20,– Best.-Nr. C 11a Kassette DM 16,–

Reddish One (10/85) 3-D-Hubschrauber Simulator (11/86) Head to Please Islanders (1/86)
Best.-Nr. C 12 Diskette DM 13,– Best.-Nr. C 12a Kassette DM 10,–

Formel One (3/86) Directory in Farbe (3/86) Reddish in the Jungle (5/86)
Best.-Nr. C 13 Diskette DM 13,– Best.-Nr. C 13a Kassette DM 10,–

Sittiegen (7/86) Jumping Jive (1/86) Eastern Night (11/86) Hardcopy C64 (11/86) Time Loader (nur für Diskette) (11/86)
Best.-Nr. C 14 Diskette DM 13,– Best.-Nr. C 14a Kassette DM 10,–

OFT-
WARE-
BESTELL-
SCHEIN

upon ausschneiden und einsenden an:

armsaka Emhald Softwareversand
ostfach 1809 7518 Bretten

Anzahl	Bestell-Nr.	Ser.-Nr. Pack	Preis

Name des Bestellers

Straße PLZ/Ort

Telefon Datum/Unterschrift

Ich wünsche folgende Bezahlung:

☐ Nachnahme (zuzügl. DM 5,10 Porto- und Versandkosten)

☐ Vorauskasse (keine Versandkosten)
Bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe ... überweisen.

ATARI®

INFOS - NEWS - BERICHTE - REVIEWS - TESTS - PROGRAMME - LISTINGS - ALLES FÜR ATARI-USER

Fröhliche Weihnachten

heißt es jetzt bald wieder. Auch dieses Jahr wird es sie geben: die Computer unter den Weihnachtsbäumen. Computerver- und -einkäufer stehen bereits jetzt voll im Streß. Wieder werden viele Computer mit falschen Vorstellungen gekauft, Laien aufgeschwatzt und an ahnungslose Grundschüler verschenkt.

Allen, die in dieser Saison ihren Computer bekommen haben und jetzt die CK in Händen halten, einen doppelten Glückwunsch. Zum Kauf dieses Heftes und – was sich aus ihrem Interesse für diese Seiten schließen läßt – vor allem zu ihrem Atari, für den wir wieder allerhand zu bieten haben. Viele Hilfsprogramme für die 8-Bit-Ataris, Maschinensprachliches von Peter Finzel, eine Basic-Tabellenkalkulation und wie immer eine bunte Mischung aus Spielen, Infos und Tricks.

Für die großen Ataris haben wir wieder einiges zusammengetragen. Zusätzlich gibt's für die großen und die kleinen Ataris demnächst noch viel mehr im neuen ATARI-magazin. Ein ausführlicher Hinweis findet sich in diesem Heft.

Bis nächstes Jahr verbleibe ich als Euer

Thomas Tausend

Atari-Leserfragen

Ich besitze einen Atari 800 XL mit Cassettenrecorder. Nun habe ich vor kurzem eine Diskettenstation 1050 geschenkt bekommen. Da ich sehr viele Spielcassetten besitze und das ewige Warten leid bin, möchte ich die Spiele nun auf Diskette kopieren. Alle bisherigen Versuche sind jedoch fehlgeschlagen – wie muß ich vorgehen?

Basic-Programme können problemlos von Cassette geladen (CLOAD) und auf Diskette gespeichert werden (SAVE "D:programme.ext"), wenn man vorher ein DOS gebootet hat. Bei Maschinenprogrammen sieht die Sache jedoch anders aus. Meistens bestehen die Programme aus mehreren Teilen und sind mit einem Kopierschutz versehen. Selbst wenn diese Daten auf Diskette kopiert werden, kann man das Programm dennoch nicht starten, ohne es auf Diskettenbetrieb abzustimmen.

1. Ich beschäftige mich zur Zeit mit der Programmierung mathematischer Funktionen in Basic. Bisweilen kommen jedoch recht sonderbare Ergebnisse dabei heraus. Gerade bei sehr großen und sehr kleinen Zahlen sind die Ergebnisse oft falsch. Was kann man da machen?

2. In diesem Zusammenhang wurde mir erklärt, daß dieses Problem mit Maschinencode gelöst werden kann. Wie kann ich in die Maschinenprogrammierung einsteigen?

3. Bei laufenden Programmen schaltet der Atari dennoch in den Farbwechselmodus um. Wie kann ich dies verhindern?

1. Obwohl der Atari sehr viel genauer rechnet als z. B. ein Apple, Commodore oder Sinclair Computer (einfacher Test: FOR N=1 TO 1000 STEP 0.1:PRINT N:NEXT N), treten auch bei den Ataris Ungenauigkeiten auf. Diese resultieren aus Rundungsfehlern, da jeder Computer mit einer festen Anzahl an Stellen rechnet. Da das Auftreten solcher Fehler sehr schlecht vorhersehbar ist, kann man sie kaum verhindern.

2. Peter Finzels Maschinensprachkurs und sein kürzlich erschienenes Buch sollen Ihnen den Einstieg in die Maschinensprache ermöglichen. Wenn Sie jedoch glauben, in MS einfacher und genauer rechnen zu können, so werden Sie eine herbe Enttäuschung erleben, da Sie jede Rechenoperation aus elementaren Additions-, Subtraktions- und Schiebebefehlen zusammensetzen müssen. (Die Betriebssystemroutinen arbeiten auch von der MS-Ebene aus nicht genauer!)

3. Mit POKE 77,0 kann dies verhindert werden. Der POKE muß allerdings mindestens einmal in ca. 7 Min. durchlaufen werden. Am besten ist es, wenn Sie ihn in Joystick-Abfragen o. ä. einbauen.

Ich habe das Programm abgetippt. Bei RUN erscheint immer ERROR 8 AT LINE xxx. Zeile xxx ist jedoch einwandfrei. Wo liegt der Fehler?

Der Fehler liegt bei ERROR 8 nicht in der Zeile, die bei der Fehlermeldung gelistet wird. ERROR 8 bedeutet, daß dem Computer beim Versuch, eine Zahl in eine Variable einzulesen (in der Regel mit READ), ein Text untergejubelt wurde. Da Zahlen in DATA-Zeilen schon als Text interpretiert werden, wenn ein Leerzeichen enthalten ist, kann dies ERROR 8 verursachen. Auch doppelte Kommas verursachen diesen Fehler. (Ein Leerstring ist ja schließlich auch keine Zahl!) Weniger häufig, aber durchaus vertreten sind Nullen, die als Buchstabe O eingegeben werden (oder l oder I statt 1). Es ist nun jedoch keineswegs notwendig, gleich das ganze Listing nochmals Zahl für Zahl zu vergleichen. Ein kleiner Trick hilft weiter. Normalerweise befindet sich der READ-Befehl innerhalb einer FOR-NEXT-Schleife oder einer GOTO-Schleife mit Vergleich und Zählvariable (wenn nicht, einfach einfügen, z. B. vor READ TEST = TEST + 1 schreiben). Bricht der Computer nach RUN mit ERROR 8 ab, so gibt TEST bzw. der aktuelle Wert der Zählvariablen (die Variable nach FOR) die Nummer des Datums aus, bei dem der Computer nicht mehr weiterkommt (einfach PRINT TEST oder eben die Schleifenvariable aus Ihrem Programm statt TEST). Sollte die Schleife (wie bei TITAN) nicht bei 0 beginnen, so ist dieser Startwert (die Zahl nach dem = in der FOR-Zeile) von dem ermittelten Wert abzuziehen.

Jetzt brauchen Sie die DATAs nur noch nachzuzählen, wobei Sie beim ersten Datum beginnen und bis zu dem ermittelten Wert weiterzählen. Hier finden Sie (+ 1) das defekte Datum. Dieser Vorgang ist so lange zu wieder-

holen, bis kein ERROR 8 mehr auftritt.

Gibt es einen Adapter, mit dem ich die Cassetten des Atari Telespiels VCS2600 verwenden kann?

Nein! Ist aber auch nicht besonders wünschenswert, wenn man einmal die Grafikqualität der beiden Geräte vergleicht!

Ich entdeckte vor kurzem eine Diskette "Ex Basic". Kann ich damit Programme des TI 99/4A verwenden?

Nein! Ex ist eine gebräuchliche Abkürzung für extended, also erweitert. Dieses Programm ist nichts anderes als ein erweitertes Atari Basic!

Ich bin Besitzer der 80-Zeichen-Karte mit Oldrunnaufstellung. Mit PRINT USR (49152) wird diese Option aktiviert. Wie kann ich nun wieder in den 40-Zeichen Modus schalten, ohne daß das Programm unterbrochen wird? Wie kann ich sonst hochauflösende Grafik und 80 Zeichen gleichzeitig betreiben?

Diese Frage gebe ich hiermit an unsere Leser weiter. Tips bitte direkt an Wolfgang Deutsch, Stephansplatz 3, 3000 Hannover 1

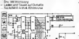
Atari ● Atari ● Atari ● Atari

Elektronische Schaltungen konstruieren mit dem ATARI

Schaltungen ... Arbeiten mit dem Joystick
Widerstand, Kondensator, Diode, Transistor, Gatter
Inverter, IC usw.
- Drucker-Hardcopy
- Laden und Saven auf Diskette
- Bauteilbibliothek 60mm usw.

ATARI
XL/XE 64K
40.– DM
inkl. Versand
Bestellung
Schein, Scheck,
Info kostenlos!
Nachnahme
+ 6.– DM

Jürgen Dörr
Einsteinstr. 8
6520 Worms 26
☎ 06241/34140



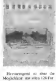
Graphic Arts Departement

Im Rausch der Farben

Der erste Eindruck war etwas enttäuschend, schon wieder ein neues Malprogramm und noch dazu im etwas blockartigen Grafikmodus 7. Nachdem die Demo-Bilder auch nicht gerade umwerfend waren, wollte ich es schon beiseite legen, doch ein Blick in die Anleitung des Programms förderte eine ganze Reihe von erstaunlichen Fähigkeiten zutage.

Das fängt bereits damit an, daß man alle 128 Farben in einem Bild verwenden kann. Welches andere Malprogramm kann das schon? "GAD" hat einen unglaublich umfangreichen Satz an Grafik-Funktionen, der alle bisherigen Programme wie "Atari Artist" oder "Micro-Painter" in den Schatten stellt. Ferner gibt es drei Editoren, mit denen Zeichensätze, Füllmuster und Pinselformen verändert werden können. Das ist aber noch nicht alles. Das Programm kann Ihre Kunstwerke auch zu Papier bringen und enthält ein Basic-Programm, mit dem "GAD"-Bilder in eigenen Programmen verwendet werden können.

Nach dem Laden des Programms (dauert sehr lange) hat man die Auswahl zwischen "Arts Gallery" und "Arts Bench". Letzteres ist das Malprogramm, zur "Gallery" kommen wir später. Zuerst die schlechte Nachricht: "Arts Bench" arbeitet ausschließlich in GRAPHICS 7, d. h. mit einer Auflösung von nur 160×96 Punkten in 4 Farben. Warum nicht die höchste Auflösung mit 160×192 Punkten gewählt wurde, bleibt unklar. Mit diesem Modus arbeiten z. B. "Atari Artist" und eine ganze Reihe anderer Malprogramme. Durch die geringe vertikale Auflösung wirken die Bilder immer etwas blockartig.



Hervorragend ist aber die Möglichkeit, mit allen 128 Farben des Atari zu arbeiten. Man verfügt über vier Farbtöpfe, die in jeder Zeile mit einem anderen Farbton gefüllt werden können. Sie wissen bestimmt (wenn Sie die Assemblerecke der letzten Ausgabe gelesen haben), daß hierbei ein wenig mit DLIs gezaubert wird. In einer horizontalen Zeile können zwar nur vier verschiedene Farben verwendet werden, im ganzen Bild sind jedoch 128 möglich. Die Farbwechsel lassen sich bequem mit dem Joystick programmieren.

Die Menge der Grafik-Funktionen von "GAD" ist schon erstaunlich. Neben den regulären Funktionen wie Linien, Rechtecke, Kreise u. a. bietet es auch ein zweistufiges Zoom, Text in vielen verschiedenen Größen und Operationen zum Verdrehen, Spiegeln und Kopieren von Bildteilen. Das ganze Bild kann in alle Richtungen verschoben werden. Ellipsen sind möglich, und auch die Koordinaten des Cursors können abgefragt oder gezielt eingegeben werden. Damit man den Überblick nicht verliert, bietet "GAD" einen Help-Screen, in dem alle Befehle zusammengefaßt sind.

Eine wirklich nützliche Einrichtung sind die drei Editoren. Da wäre zunächst ein Font-Editor, mit dem Zeichensätze geladen, verändert und gespeichert werden können. Weiterhin gibt es einen Brush-Editor, er dient zum Verändern der Pinselform. Ein ganzer Satz von Pinseln kann auf Diskette gespeichert und bei Bedarf nachgeladen werden. Mit dem dritten Editor werden Füllmuster verändert und gefüllte Kreise und Rechtecke gezeichnet. Die Editoren sind reichlich mit Hilfsfunktionen ausgestattet.

Der Programmteil "Arts Gallery" dient zum Anzeigen und Ausdrucken von "GAD"-Bildern und kann auf eigene Disketten kopiert werden. Die Option zum Ausdrucken der Bilder funktioniert dann problemlos, wenn man einen der im Programm gespeicherten Drucker besitzt (Epson etc.). Andere Drucker können mit einem Menü angepaßt werden, allerdings sollte man sich dazu in seinem Drucker-Handbuch etwas auskennen. Grundvoraussetzung ist, daß Ihr Drucker einen Modus mit 960 Punkten pro Zeile besitzt. Damit dürften sich 1029-Besitzer schon auf dem Abstellgleis befinden. Der Ausdruck ist nur in einer einzigen Größe (ca. DIN A5) möglich. Farben werden als Graustufen dargestellt (wählbar).

Sie finden auch ein kleines Maschinenprogramm auf der Diskette, mit dem die Bilder in eigenen Programmen geladen werden können. Für Assembler-Programmierer ist sogar der Quellcode enthalten.

Die englischsprachige Anleitung von "Graphic Arts Departement" umfaßt 8 Seiten und enthält alle wesentlichen Informationen. Sieht man von der geringen Auflösung ab, dann überzeugt "GAD" durch die Unmenge an Funktionen.

Preis ca. 70 DM Disk
System
alle Atari Computer ab 48 K

Weitere Informationen:
Compy Shop
Gneisenaustr. 29
4330 Mülheim

Peter Finzel

The Music-Studio

Mit diesem Programm bietet Ihnen Activision ein komplettes Tonstudio auf Diskette an. Ihr Atari wird zum elektronischen Notenblatt und Orchester. Sie müssen nur die Noten eingeben und die Instrumente auswählen, schon tönt die Melodie aus dem Lautsprecher. Wer mit Noten auf Kriegsfuß steht, kann seine Musik sogar mit der "Music-Paintbox" "malen". Einfacher geht's wirklich nicht mehr.

Das "Music Studio" besteht aus drei Teilen: dem "Music Editor", dem "Sound Engineering Room" und der "Music-Paintbox". Angenehmerweise befinden sich alle drei Teile permanent im Speicher, so daß nichts von Diskette nachgeladen werden muß.

Music-Editor

Die Zentrale von "Music Studio" bildet der Editor. Der Bildschirm besteht aus zwei großen Notenzeilen (eine mit Violin-, eine mit Baßschlüssel) und einem Menü im unteren Teil. Die Auswahl geschieht mit dem Joystick (Symbol ansteuern und Knopf drücken). Ebenso einfach werden die Noten eingegeben, indem man Notenlänge und Instrument anwählt und die Note ein- oder mehrmals in der Notenzeile ablegt. Der Editor macht seinem Namen alle Ehre und erlaubt es, Noten zu löschen, einzusetzen und zu kopieren. Freilich lassen sich auch Pausen, Vorzeichen und Punktierungen einsetzen, ebenso kann die Tonart frei gewählt werden.

Beim "Music Studio" ist es möglich, bis zu 15 Instrumente in einem Musikstück einzusetzen. Merkwürdigerweise können nur drei Instrumente gleichzeitig ertönen, obwohl der Atari doch über vier Tonkanäle verfügt. Sollte das vielleicht mit der Umsetzung vom C64 zusammenhängen? Lob verdient aber die Auswahl der Instrumente, die durch den Farbcode der Noten bestimmt wird. Auf diese Weise ist man nicht auf ein Instrument pro Stimme festgelegt, sondern kann praktisch mit jeder Note das Instrument wechseln. Auch bis zu 3 Textzeilen können eingegeben werden, die dann beim Abspielen mit den Noten über den Bildschirm laufen.

Tontechnik

Im "Sound Engineering Room" können Sie nach Herzenslust mit den akustischen Möglichkeiten des Atari experimentieren. Acht Schieberegler verändern die ADSR-Charakteristik der 15 Instrumente, spezielle Verzerrungsmodi des POKEY-Chips erzeugen z.B. Schlagzeugeffekte. Die 15 Instrumente entsprechen dann direkt den Farbcodierungen im "Music Editor".

Paintbox

Wer mit Noten, Vorzeichen und Pausen nichts anzufangen weiß, kann die Musik auch über die "Paintbox" erzeugen. Anstelle von Noten werden nur verschieden lange, farbige Balken auf die Zeile gemalt. Je länger der Balken, desto länger der Ton. Durch Druck auf den Joystick kann diese Darstellung in normale Noten umgewandelt und mit dem "Music Editor" weiter bearbeitet werden.

"The Music Studio" wird auf einer Diskette mit den Versionen für Atari und C64 geliefert. Das englischsprachige Handbuch ist mit fast 60 Seiten recht ausführlich. Merkwürdig ist jedoch, daß eine Funktion nicht für den Atari übernommen wurde, die das Programm sicherlich noch aufwerten würde: gemeint ist der Ausdruck. Alle anderen Versionen von "Music Studio" können die Noten zu Papier bringen.

System Atari ab 48 K
Preis ca. 69,– DM D

Weitere Informationen
Compy-Shop
Gneisenaustr. 29
4330 Mülheim

Peter Finzel

Neues für den Gabentisch

Die Tage werden kürzer, die Nächte länger, die Zeit für Computerspiele bricht an. Gerade rechtzeitig zum Weihnachtsfest kommen zwei neue Spiele vom AMC-Verlag sowie ein neu veröffentlichter Klassiker von Databyte.

Bilbo

Kennen Sie noch den Spielautomaten "Lady Bug"? Wenn Ihnen dieses an den Klassiker "Pacman" angelehnte Spiel Freude macht, dann ist "Bilbo" genau das Richtige für Sie. Der mit dem Joystick gesteuerte Zwerg rast durch ein Labyrinth, sammelt Punkte und wird dabei – wie nicht anders zu erwarten – von Geistern verfolgt. Das Spiel erhält seinen Reiz durch eine Anzahl von Drehtüren, die der Zwerg bei seiner Reise umlegen kann, die Geister jedoch nicht. Sammelt man die Buchstaben EXTRA (in dieser Reihenfolge!), dann läßt ein Multiplikator den Score in die Höhe schnellen.

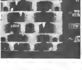
Unsere Beta-Version hatte zwar noch nicht alle Features, aber man kann schon jetzt sagen, daß das Spiel grafisch gut gestaltet und technisch perfekt programmiert ist. Nur die Spielidee ist schon etwas abgenutzt. Fällt den deutschen Programmierern außer "Pacman" und Konsorten nichts mehr Neues ein?

Wer Labyrinth-Spiele mag, wird mit "Bilbo" dennoch zufrieden sein. Der Preis für die Diskette ist mit ca. 20,– DM jedenfalls sehr günstig.

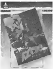
Tales of Dragons and Cavemen

Auch hinter diesem etwas langen Titel verbirgt sich ein neues Spiel vom AMC-Verlag. Es geht über vier Bildschirme, in denen man mit einem Seil von Stock zu Stock klettert, um einen wertvollen Vogel zu retten ("Donkey Kong" läßt grüßen). Natürlich ist es fast unvermeidlich, daß einen dabei herabfallende Eier (!) und Steine behindern.

Leider lag uns noch keine testfähige Version des Spiels vor. Die Bilder, die wir erhielten, machen allerdings keinen schlechten Eindruck. Auch dieses Spiel kostet rund 20,– DM auf Diskette.



Die Spiele Tales of Dragon und Cavemen aus dem AMC Verlag



Aztec

Dieses Programm ist ein echter Klassiker unter den Computerspielen. „Aztec" war einer der allerersten Action-Adventures (allerdings nur für Apple-II-Computer). Nun gibt es eine Version für Atari-Computer, die nicht mit Tasten – wie beim Original –, sondern per Joystick gesteuert wird. Die großformatige Spielfigur, die stark an Indiana Jones erinnert, kann laufen, rennen, kriechen, springen und noch einiges mehr. Gegenspieler sind einige recht gut animierte Monster wie Saurier, Löwen und Schlangen.

Ziel des Spiels ist es, in eine alte Pyramide einzudringen und den dort versteckten Goldschatz zutage zu fördern. Viele Hilfsmittel findet man in der Pyramide, wenn man fleißig die Knochenhaufen der zuvor gescheiterten Abenteurer durchsucht. Da das Spiel schon einige Jahre alt ist, sind Grafik und Bedienung teilweise nicht so ausgereift, wie man es heute gewohnt ist. Wer jedoch einen echten Klassiker in seine Software-Sammlung einreihen will, sollte bei „Aztec" zum Preis von ca. 60,– DM zugreifen.

Weitere Informationen:
AMC-Verlag
Blücherstr. 17
6200 Wiesbaden

Compy-Shop
Gneisenaustr. 29
4330 Mülheim

Peter [illegible]

Mercenary – Escape from Targ

Paul Woakes, der Autor von „Encounter", hat wieder einmal zugeschlagen! Für sein neues Werk „Mercenary" hat er sich reichlich Zeit gelassen. Doch das Warten auf dieses sehr komplexe Programm, eine Mischung aus Flugsimulator, Action und Adventure, hat sich gelohnt.

Sie schlüpfen in die Rolle eines Raumfahrers im 21. Jahrhundert, der aufgrund eines Maschinenschadens auf dem Planeten Targ notlanden muß. Während Sie unverletzt bleiben, wird das Raumschiff bis auf den Bordcomputer Benson, der Sie mit wichtigen Informationen versorgt, völlig zerstört. Die Ureinwohner des Planeten, die Palyars, kämpfen gegen die Mechanoids, eine übermächtige Roboter-Rasse, die den Planeten bevölkert. Sie mußten sich in eine Raumfestung, die in einer Umlaufbahn um den Planeten kreist, zurückziehen.

Die Aufgabe des Spielers besteht nun darin, genügend Geld für ein neues Raumschiff zu verdienen, mit dem man von Targ entkommen kann. Zu Beginn steht Ihnen ein Gleiter zur Verfügung, mit dem Sie den Planeten erkunden können. Man erkennt eine Menge Gebäude, Brücken, Radarpisten und sogar Reklameschilder.

Die Grafik der getesteten Atari-Version fasziniert durch die Vielzahl von Objekten, die alle in hochauflösender Vektor-Grafik dargestellt sind und sich flackerfrei mit beeindruckender Geschwindigkeit aufbauen. Es ist schon ein Erlebnis, wenn man z.B. über eine Brücke fliegt, hinabstürzt und schließlich unter ihr hindurchgleitet. „Elite" wirkt gegen diese schnellen 3-D-Grafik-Routinen wie ein Basic-Programm.

Aber damit nicht genug! Wenn Sie bestimmte Gebäude betreten, können Sie per Aufzug die Unterwelt Targs besuchen. In den unzähligen Zimmern und Korridoren finden Sie eine Reihe von nützlichen Gegenständen. Zehn davon darf man mitnehmen. Über den Bordcomputer Benson erhalten Sie auch Job-Angebote wie das Transportieren eines bestimmten Gegenstands zur Raumfestung, oder Sie verbünden sich mit den Palyars, um ihnen im Kampf gegen die Mechanoids zu helfen. Bei erfolgreicher Mission erhöht sich Ihr Geldkonto.

Zum Glück läßt sich der Spielstand speichern, so daß man zu einem späteren Zeitpunkt weiterspielen kann. Dies ist auch bitter nötig, denn Raumschiffe sind bekanntlich sehr teuer. Noch einen kleinen Gag bietet die Atari-Version: Schießt man das Commodore-Reklameschild ab, meldet Benson fröhlich: „Good Shoot".

Dieses Spitzenprogramm sollte in der Software-Sammlung von Atari-Fans, die in letzter Zeit nicht gerade mit guten Programmen überhäuft wurden, nicht fehlen.

System	Atari Computer C 64, C 128
Preis	ca. 49,– DM Disk ca. 39,– DM Cass.

Bernd Müller

ATARI • ATARI • ATARI • ATARI • ATARI • ATARI • ATARI • ATARI

H.G. Dreeser, Soft- und Hardware

Wir bekommen laufend die aktuellsten Produkte für den Atari 600/800/130 XL/XE sowie Atari ST. Nutzen Sie unseren Telefon- und Auftragsservice zu den angegebenen Zeiten, damit auch Sie über die Neuheiten informiert sind.

Atari-Cassetten-Software		Atari-ST-Software	
Kick Start	[illegible] DM	Fire Blaster	[illegible] DM
Greatest Hits 1	21,90 DM	Warzone	[illegible] DM
Smash Hits 5	[illegible] DM	Pinball Factory	71,00 DM
Handball	[illegible] DM	[illegible]	[illegible] DM
Comet Alert	24,90 DM	Strip Poker	[illegible] DM
Airline	[illegible] DM	Electronic Pool	34,90 DM
Earthquake	[illegible] DM	[illegible]	[illegible] DM
Atari-Disketten-Software		Schach (Dt.)	[illegible] DM
Flight Sim. II	[illegible] DM	Shuttle II	[illegible] DM
Gun Star	42,90 DM	[illegible]	[illegible] DM
Leaderboard	[illegible] DM	Space Battle	[illegible] DM
Hard Ball	57,90 DM	[illegible]	[illegible] DM
Silent Service	[illegible] DM	[illegible] (Dt.)	77,90 DM
Smash Hits 5	[illegible] DM	[illegible] (Dt.)	[illegible] DM
Music Studio	77,90 DM	ST Toolkit (Dt.)	[illegible] DM
		K-Graph (Dt.)	[illegible] DM

Fordern Sie unsere [illegible]
Dreeser Soft- u. Hardware [illegible]
[illegible]

ATARI • ATARI • ATARI • ATARI • ATARI • ATARI • ATARI • ATARI

AMC
VERLAG – WIESBADEN
Armin Stürmer Blücherstr. 17, 6200 Wiesbaden

Software für den gehobenen Geschmack
ATARI 800 XL / 600 XL (64K) / 130 XE

Erhältlich

PYRAMIDOS	AMC	DM 29,–
MIKE s Slotmachine *	AMC	DM 19,–
THE SOUNDMACHINE	PFP	DM 29,80
DESIGNMASTER	PFP	DM 19,80

Neu im Programm

BILBO*	AMC	DM 19,–
TALES OF DRAGONS AND CAVEMEN*	AMC	DM 19,–

(* 3er Pack DM 50,–)

Und alle 8 Wochen neu

AMC-SOFT	AMC	DM 8,–

Das Disk-Magazin mit Game
(Jahresabo (6 Ausgaben) statt DM 48,– nur DM 40,–)

Nachnahme (zuzügl. DM 5,– Versandkosten)
Vorauskasse (keine Versandkosten)

Händleranfragen erwünscht!
INFO KOSTENLOS ANFORDERN!

Atari-Maschinensprache für Anfänger

Das Assemblerbuch von Peter Finzel ist ein Kurs zur Einführung in die 6502-Maschinensprache, passend zum Atmas-II Makroassembler und speziell auf Atari-Computer zugeschnitten. Es werden keinerlei Vorkenntnisse in Assembler erwartet.

In drei großen Abschnitten wird Ihnen die Welt der Maschinenprogrammierung systematisch nähergebracht. Der erste Teil beschäftigt sich mit den Grundlagen wie Binär- und Hexadezimalsystem und dem Aufbau des 6502-Prozessors. Dort erfahren Sie auch, was eine Direktive, ein Opcode und eine Adressierungsart ist.

Im zweiten Teil erwartet Sie der Befehlssatz des 6502-Prozessors. Viele Beispiele, die Sie gleich mit Atmas II eintippen und ausprobieren können, zeigen die Befehle im praktischen Einsatz.

Leider nützt Ihnen die Kenntnis der 6502-Befehle wenig, wenn Sie mit den speziellen Eigenschaften des Ataris wie Custom-Chips, Interrupts und dem Betriebssystem nicht vertraut sind. Aus diesem Grund wurde der dritte Teil allein diesen Anwendungen gewidmet. Hier finden Sie ein Beispiel zur Ein-/Ausgabeprogrammierung, zur Player-Missile-Grafik und nicht zuletzt zwei Beispiele zur Interrupt-Technik.

Zu allen Listings gibt es eine ganze Menge an Hintergrundinformationen, die Sie bei eigenen Assembler-Projekten unterstützen. Nebenbei erhalten Sie noch eine Fülle praktischer Tips, z.B. wie man Assembler-Programme mit Basic koppelt, wie man selbststartende Programme schreibt und vieles mehr. Ein alphabetisch geordnetes Assembler-Vokabular im Anhang des Buches wird Ihnen auch beim Lesen von Fachartikeln eine Hilfe sein, wenn Sie mit Begriffen wie LSB, Zero-Page oder DMA noch nicht vertraut sind.

Das Assemblerbuch umfaßt 169 Seiten im Format DIN A5 mit abwaschbarem Umschlag. Es kann zum Preis von 29,80 DM direkt über den Verlag Computer Kontakt bezogen werden. Bestellschein siehe "Take 9".

Oskar Schlemmer

6502-Maschinensprache

Ein Kurs für Atari User – 3. Folge

Diesmal wird es ernst, wir beschäftigen uns mit dem ersten Teil des Befehlssatzes des 6502-Prozessors und bringen erste Programmbeispiele dazu. In der letzten Folge sind wir gerade bis zum LDA-Befehl vorgedrungen und haben uns die wichtigsten Adressierungsarten näher angesehen.

Bevor wir nun mit der Besprechung des Befehlssatzes fortfahren, sollten wir uns noch mit einer wichtigen Einrichtung des 6502 vertraut machen, den Prozessor-Flags. Sie zeigen besondere Zustände an (wenn z.B. bei einem Rechenvorgang der Zahlenbereich überschritten wurde). Abhängig von diesen Flags kann der Programmablauf beeinflußt werden. Hier die Bedeutung der sieben Flags:

Z: Das Zero-Flag zeigt an, daß das Ergebnis der letzten Operation null war. Es muß sich dabei nicht um eine Rechenoperation handeln; auch wenn eine Null durch einen Ladevorgang in ein Register kommt, wird das Z-Flag gesetzt.

N: Das Negative-Flag zeigt an, wenn das Ergebnis der letzten Operation als negativ interpretiert werden kann. (Das Bit mit dem höchsten Wert ist gesetzt.)

C: Das Carry-Flag wird gesetzt, wenn ein Überlauf bei einem Rechenvorgang aufgetreten ist. Die größte Zahl, die der 6502 im Akku darstellen kann, ist 255. Sollte das Ergebnis einer Rechnung größer sein, so wird das C-Flag gesetzt. In vielen Fällen kann man sich dieses Flag auch als „neuntes Bit" des Akkus vorstellen.

V: Das Overflow-Flag zeigt an, ob bei Rechenoperationen mit vorzeichenbehafteten Zahlen ein Überlauf aufgetreten ist. Stellen Sie sich das bitte so vor: Das achte Bit gibt bei einer derartigen Zahl ihr Vorzeichen an. Findet bei einer Addition ein Überlauf vom siebten ins achte Bit statt, so würde das Vorzeichen verfälscht. Das Overflow-Flag wäre in diesem Fall gesetzt und zeigt an, daß das Vorzeichen nicht mehr stimmt.

B: Das Break-Flag wird gesetzt, wenn ein BRK-Befehl ausgeführt wird.

I: Das Interrupt-Disable-Flag kann eine bestimmte Gruppe von Interrupts (die IRQs) verhindern. Es empfiehlt sich allerdings beim Atari, dieses Flag nur zu verwenden, wenn man sich mit Interrupts auskennt. Setzt man dieses Flag, so nimmt der Computer z.B. keine Tastatureingaben mehr an!

D: Mit dem Dezimal-Flag läßt sich der Prozessor in einen dezimalen Rechenmodus umschalten. Er benutzt dann den sog. BCD-Code, mit dem in einer 8-Bit-Zahl zwei dezimale Ziffern untergebracht werden können. Die größte Zahl ist 99 (dargestellt als $99). Sollte das Ergebnis eines Rechenvorgangs größer sein, so wird das C-Flag gesetzt.

Sie haben vielleicht bemerkt, daß die beiden letzten Flags eine gewisse Sonderstellung einnehmen, da sie keine Zustände anzeigen, sondern gewisse Eigenschaften des Prozessors umschalten.

Bei der nun folgenden Vorstellung der Befehle wird immer mitangegeben, welches Flag sie beeinflussen.

Laden und Speichern

Die zweifellos am häufigsten verwendeten Befehle dienen zum Laden und Speichern des Akkumulators. Den LDA-Befehl kennen Sie bereits aus der letzten Folge; er beeinflußt das N- und das Z-Flag. Die Umkehrung davon heißt STA (Store Accumulator) und dient zum Ablegen des Akkuinhalts im Speicher. Flags werden nicht verändert. Die zulässigen Adressierungsarten können Sie (wie auch bei den folgenden Befehlen) aus Tabelle 1 entnehmen. Eine Kombination von Adressierungsart und Befehl ist erlaubt, wenn am Kreuzungspunkt ein Code in der Tabelle eingetragen ist. Die Codes brauchen Sie sich übrigens nicht zu merken, sie werden vom Assembler automatisch erzeugt.

Indexregister

Wie für den Akku gibt es auch Lade- und Speicherbefehle für die beiden Indexregister X und Y. Zum Laden des X-Registers dient der Befehl LDX, gespeichert wird es mit STX. Analog heißen die auf das Y-Register wirkenden Befehle LDY bzw. STY. Im Gegensatz zu den auf den Akku bezogenen Befehlen sind hier keine indirekten Adressierungsarten zulässig (s. Tabelle 1). Während LDX und LDY wiederum das Z- und das N-Flag beeinflussen, lassen die Speicherbefehle die Flags unverändert.

Mit diesen wenigen Befehlen (und einigen Assembler-Direktiven aus der zweiten Folge) kann man schon ein erstes einfaches Programm zusammenstellen. In Listing 1 verwenden wir die besprochenen Lade-

und Speicherbefehle um drei Farbregister mit neuen Werten zu besetzen. Die Werte werden zunächst per Immediate Adressierung in die Prozessor Register geladen und anschließend mit absolut adressierten Store-Befehlen in die Schattenregister der Farben geschrieben. Die EQU-Direktiven am Anfang des Programms ordnen den Adressen der Schattenregister symbolische Namen zu, um das Ganze verständlicher zu machen. Durch die ORG-Anweisung wird das Programm an die Adresse $A800 assembliert.

Das Programm ist für "Atmas-II" geschrieben, kann aber auch für andere Assembler modifiziert werden. Zum Starten des Programms gehen Sie in den Monitor (bei Atmas mit Control P) und benutzen die GOTO-Funktion zur Adresse $A800 ($-Zeichen nicht eingeben).

Den letzten Befehl des Programms haben wir bisher noch nicht besprochen. Wir werden das auf später verschieben, da seine Wirkungsweise recht kompliziert ist. Merken Sie sich nur, daß er BRK (für Break) heißt und die Rückkehr zum Monitor zusammen mit einer Anzeige der Prozessor-Register bewirkt.

Transferbefehle

Eine weitere Gruppe von Befehlen erlaubt den Austausch von Informationen zwischen den Registern. Die Liste der Transferbefehle lautet:

TAX Der Akku wird ins X-Register kopiert.

TXA Das X-Register wird in den Akku kopiert.

TAY Der Akku wird ins Y-Register kopiert.

TYA Das Y-Register wird in den Akku kopiert.

TSX Dies ist der einzige Befehl, mit dem der Stapelzeiger gelesen werden kann. Sein Inhalt wird ins X-Register kopiert.

TXS Mit diesem Befehl kann der Stapelzeiger mit einem neuen Wert (dem Inhalt des X-Registers) überschrieben werden.

Alle Transferbefehle außer TXS beeinflussen das N- und das Z-Flag. TXS verändert keine Flags. Hier wird eine Adressierungsart verwendet, die wir in der letzten Folge noch nicht besprochen haben. Sie nennt sich Implied (implizit) und bedeutet nichts anderes, als daß der Befehl keinen Operanden benötigt und daher nur ein Byte lang ist.

Zählbefehle

Eine der hervorstechendsten Eigenschaften der beiden Index-Register ist, daß Sie problemlos auf- und abwärts zählen können. Dazu dienen die folgenden vier Befehle:

INX Increment X-Register. Zum Inhalt des X-Registers wird eine Eins addiert. Mehr oder minder entspricht dies der in Basic oft benutzten Form X = X+1.

DEX Vom X-Register wird eine Eins subtrahiert.

INY Zum Y-Register wird eine Eins addiert.

DEY Vom Y-Register wird eine Eins subtrahiert.

Alle vier Befehle beeinflussen wieder das Z- und das N-Flag und benutzen die implizite Adressierung. Sie werden später noch sehen, daß man mit diesen Befehlen und Anweisungen zum Abfragen der Flags bequem Schleifen programmieren kann. Während diese Befehle nur auf die Index-Register wirken, gibt es noch zwei weitere, die beliebige Speicherzellen herauf- und herabzählen können.

INC Increment Memory. Addiert eine Eins zu einer beliebigen Speicherzelle.

DEC Decrement Memory. Subtrahiert eine Eins von einer beliebigen Speicherzelle.

Die Adressierungsarten können Sie Tabelle 1 entnehmen. Auch diese beiden Befehle wirken auf das N- und das Z-Flag.

Addition und Subtraktion

Eine der wichtigsten Eigenschaften eines Mikroprozessors ist sicherlich die Fähigkeit zu rechnen. Leider ist der 6502 (wie die meisten 8-Bit-Prozessoren) auf diesem Gebiet nicht eben opulent ausgestattet; er kennt nur Addition und Subtraktion. Alles andere (z. B. Division, Multiplikation) muß man selbst programmieren.

ADC Add with Carry. Ein Byte aus dem Speicher wird zum Akku-Inhalt addiert. Da auch das Carry-Flag dazugezählt wird, kann man auf diese Weise mehrfach genaue Additionen ausführen. Das Ergebnis der Addition bleibt im Akku zurück. Wurde der Zahlenbereich überschritten, so ist nach dem ADC-Befehl das Carry-Flag gesetzt. Vor jeder Addition sollte das Carry-Flag zurückgesetzt werden.

SBC Subtract with Borrow. Der Befehl subtrahiert eine Speicherzelle vom Akkumulator und berücksichtigt einen Überlauf von vorhergehenden Rechenoperationen. Auch hier wird das Carry-Flag als Merker für einen Übertrag benutzt, nur diesmal in umgekehrter Bedeutung. Das C-Flag ist nämlich beim Auftreten eines Übertrags zum nächsten Byte gleich null. Aus diesem Grund muß das C-Flag auch vor einem Subtraktionsbefehl gesetzt werden.

ADC und SBC beeinflussen N, Z, C und auch das V-Flag.

Verändern der Flags

Bei der Besprechung der letzten beiden Befehle haben wir gesehen, daß es möglich sein muß, die Flags gezielt zu setzen oder zurückzusetzen. Zu diesem Zweck gibt es eine Reihe von Befehlen.

SEC Set Carry Flag. Das C-Flag wird auf eins gesetzt.

CLD Clear Decimal Mode. Der Befehl schaltet wieder in den binären Rechenmodus zurück (D-Flag gleich null).

CLC Clear Carry. Das C-Flag wird auf null gesetzt.

SED Set Decimal Mode. Das D-Flag wird gesetzt und bringt den Prozessor somit in die dezimale Rechenweise.

Befehl	Implied	Immediate	Zero-Page	Zero-Page,X	Zero-Page,Y	Absolut	Absolut,X	Absolut,Y	(Indirekt,X)	(Indirekt),Y
ADC		69	65	75		6D	7D	79	61	71
CLC	18									
CLD	D8									
CLI	58									
CLV	B8									
DEC			C6	D6		CE	DE			
DEX	CA									
DEY	88									
INC			E6	F6		EE	FE			
INX	E8									
INY	C8									
LDA		A9	A5	B5		AD	BD	B9	A1	B1
LDX		A2	A6		B6	AE		BE		
LDY		A0	A4	B4		AC	BC			
SBC		E9	E5	F5		ED	FD	F9	E1	F1
SEC	38									
SED	F8									
SEI	78									
STA			85	95		8D	9D	99	81	91
STX			86		96	8E				
STY			84	94		8C				
TAX	AA									
TAY	A8									
TYA	98									
TSX	BA									
TXA	8A									
TXS	9A									

Tabelle 1: 6502-Befehle, Teil 1

SEI Set Interrupt Disable. Das I-Flag wird gesetzt und verriegelt eine Gruppe von Interrupts. Da Interrupts erst Thema einer späteren Folge sind, soll die nähere Besprechung dieses Befehls noch ein wenig verschoben werden.

CLI Clear Interrupt Flag. Die IRQ-Interrupts werden wieder freigegeben.

CLV Clear Overflow. Das Overflow-Flag wird gelöscht. Es gibt keinen ausdrücklichen Befehl zum Setzen dieses Flags.

Alle Befehle der zuletzt besprochenen Gruppe benutzen ausschließlich die implizite Adressierungsart und sind daher 1-Byte-Befehle.

Beispiel

Da alle Theorie bekanntlich grau ist, wollen wir uns nun ein Beispiel ansehen, in dem einige der zuvor besprochenen Befehle vorkommen. Listing 2 zeigt ein Assembler-Programm, das einige Zeichen am Bildschirm ausgibt. Da wir uns mit Unterprogrammen und Betriebssystem noch nicht beschäftigt haben, werden wir den Bildschirm-Code der Zeichen einfach in den Video-Speicher schreiben. Dieser Speicherabschnitt umfaßt 960 Byte, die jeweils ein Zeichen auf dem Bildschirm festlegen (jedenfalls in GRAPHICS 0). Das erste Byte bestimmt das Zeichen in der linken oberen Ecke, die 39 folgenden Byte bilden die erste Zeile. Danach schließen sich wieder 40 Byte für die zweite Zeile an usw.

Da der Videospeicher beim Atari dank des Videoprozessors Antic nicht an einen bestimmten Speicherbereich gebunden ist, müssen wir uns seine Anfangsadresse besorgen. Wir finden sie in den Speicherzellen $58 und $59 der Zero-Page, die gewöhnlich mit dem Label SAVMSC bezeichnet werden. Deshalb befindet sich am Anfang des Listings eine Definition mit einer entsprechenden EQU-Direktive. Natürlich könnten wir auch die Zahlenwerte benutzen, aber denken Sie immer daran, daß ein Listing so verständlich wie nur möglich sein sollte. Mit dem zweiten EQU wird eine Variable in der Zero-Page definiert, die wir benötigen, um die indirekte Adressierungsart einsetzen zu können.

Das Programm soll erlauben, daß die Zeichen an einer beliebigen Stelle im Video-Speicher abgelegt werden. Daher wird zur Anfangsadresse in SAVMSC eine Konstante (die auch per EQU definiert wurde) addiert und das Ergebnis in ZEIGER abgelegt. Nehmen wir an, die Zeichen sollen ungefähr in der Mitte des Bildschirms auftauchen, so kann man z.B. 494 addieren. Mit mehreren indirekt indizierten STA-Befehlen werden die Zeichen ab dieser Stelle ins Video-RAM geschrieben, die Weiterschaltung auf das nächste Zeichen erfolgt mit dem Y-Register, das zwischen zwei STA-Befehlen erhöht wird.

Interessant ist auch die Addition der Konstanten zu SAVMSC. Da es sich um 2-Byte-Werte handelt, ist eine mehrfach genaue Addition nötig. Das C-Flag wird auf Null gesetzt, dann werden zuerst die LSBs (die niederwertigen Bytes) addiert. Sollte dabei ein Überlauf auftreten, so bleibt dieser im Carry-Flag und wird bei der anschließenden Addition der MSBs (der höherwertigen Bytes) berücksichtigt. Deshalb darf vor dem zweiten ADC-Befehl kein CLC erfolgen! Die Zerlegung der Konstanten ABSTAND in LSB und MSB kann beim ATMAS-II recht einfach erfolgen. Man schreibt ABSTAND:L für das LSB bzw. ABSTAND:H für das MSB.

In der nächsten Folge werden wir die restlichen 6502-Befehle besprechen und uns dabei einige weitere (dann schon etwas kompliziertere) Befehle ansehen.

Peter Finzel

Ihr direkter Draht zur Redaktion:
Telefon 0 72 52 / 42 9 48
Für Briefe:
Computer Kontakt
Postfach 16 40
7518 Bretten

An alle Atari-User!

Wenn Sie dieses Heft bereits von hinten betrachtet haben, dann sind Sie schon informiert: Neben der CK wird demnächst noch viel mehr für die Atari-Freunde aus unserem Verlag kommen: das ATARI magazin. Für User der kleinen 8-Bit-Ataris wie auch für die Besitzer der Atari-ST-Computer wird das neue Magazin eine geballte Ladung an Information bringen.

Geboten wird aber noch mehr: Die Besitzer von Atari-Computern aller Größen sind aufgerufen, durch Programm-Einsendungen am neuen ATARI magazin mitzuwirken. Auch Sie haben die Möglichkeit, zum ständigen freien Mitarbeiter zu werden und so Ihrem Hobby eine lukrative Seite abzugewinnen. Sind Sie interessiert? Dann schreiben Sie uns Name, Adresse und Telefonnummer. Wir setzen uns mit Ihnen in Verbindung.

ATARI magazin
Redaktion
Postfach 1640
7518 Bretten

Listing 1

```
*
* Drei Farbregister laden
*
COLOR1  EQU $2C5
COLOR2  EQU $2C6
COLOR4  EQU $2C8
*
        ORG $A000
*
        LDA #$0A      hellgrau
        LDX #$00      schwarz
        LDY #$A8      hellgruen
        STA COLOR4    Rahmen
        STX COLOR1    Schrift
        STY COLOR2    Hintergrund
        BRK
```

Listing 2

```
*
* Einfache Textausgabe
*
SAVMSC  EQU $58   Zeiger auf Video-Ram
ZEIGER  EQU $D4   Variable in Zero-Page
ABSTAND EQU 494   Abstand zur Bildmitte
*
        ORG $A000
*
        CLC              Carry auf 0
        LDA SAVMSC       erst die LSBs
        ADC #ABSTAND:L   addieren
        STA ZEIGER       und speichern
        LDA SAVMSC+1     jetzt die MSBs
        ADC #ABSTAND:H   addieren
        STA ZEIGER+1
        LDY #0           Offset=0
        LDA #$23         erstes Zeichen
        STA (ZEIGER),Y
        INY              Offset=1
        LDA #$30         naechstes Zeichen
        STA (ZEIGER),Y
        INY              Offset=2..
        LDA #$35         letztes Zeichen
        STA (ZEIGER),Y
        BRK
```

Atari **COMPY SHOP** **Atari**

FÜR DEN COMPUTER ZUHAUS

16-K-Bibomon

Der 16-K-Bibomon ist ein Maschinensprach-Monitor, der fest in Ihrem Atari-Computer der Serien 800XL und 130XE eingebaut wird. Sie erhalten mit diesem Monitor eine Centronics-Parallel-Schnittstelle und eine Hardcopy-Routine in Ihrem Computer. Diese Hardcopy-Routine können Sie **JEDERZEIT** durch einen Tastendruck auslösen. 4 verschiedene Drucker sind in Anwendung der Hardcopy-Routine enthalten!
Ebenfalls fest im Computer eingebaut wird ein DOS. Sie brauchen also kein DOS mehr zu laden, um Programme zu speichern oder zu laden!
Auch den Monitor erreichen Sie jederzeit durch einen Tastendruck. Sie können mit diesem Monitor Speicherbereiche: Disassemblieren, Listen, Vergleichen, Verschieben, Absuchen. Sie können Sektoren einlesen, verändern, schreiben. Der Monitor verfügt über einen leistungsfähigen Zeilenassembler. Und vieles mehr.
Mit dem Monitor bekommen Sie ein ausführliches, 70seitiges, deutsches Handbuch.
Der 16-K-Bibomon arbeitet mit der Speedy 1050 zusammen!

Das alles bekommen Sie exklusiv bei uns — im **COMPY-SHOP!**

16-K-Bibomon, für Atari 800XL	448 — DM
16-K-Bibomon, für Atari 130XE	DM 448 — DM
Einbauen des 16-K-Bibomon in Ihren Computer zzgl.	60 — DM
Atari 800XL 320 KRAM + 16-K-Bibomon	798 — DM
Atari 130XE 192 KRAM + 16-K-Bibomon	849 — DM

— Testbericht in der COMPUTER KONTAKT 8/9-85 —

FORDERN SIE UNSEREN AUSFÜHRLICHEN PROSPEKT AN! HÄNDLER-ANFRAGEN ERWÜNSCHT!

Speedy 1050

Speedy 1050 ist ein Hardware-Zusatz, der in Ihr Atari 1050-Laufwerk eingesteckt wird. Sie haben dann sofort mehrere Vorteile:

- **— Drei Speicherdichten**
  - **1. 88 KByte Single-Density**
  - **2. 127 KByte Medium-Density**
  - **3. 176 KByte Double-Density**
- **— Drei Übertragungs-Geschwindigkeiten**
  - **1. Atari 1050 Normal — ca. 19 200 baud**
  - **2. Speedy 1050 Normal — ca. 25 400 Baud**
  - **3. Speedy 1050 Super-Speed — ca. 56 200 Baud**
- **— Die Speedy 1050 ist voll programmierbar**
- **— Formatiert jedes der drei Formate in 9 Sekunden**
- **— Leichter Einbau, von jedem Laien ausführbar**
- **— Benutzerfreundliche Software**
- **— Arbeitet mit dem 16-K-Bibomon zusammen**
- **— Kann kopiergeschützte Software kopieren**
- **— Digitale Trackanzeige (Speedy T)**
- **— Akustischer Fehlermelder (Speedy T)**

WICHTIG! SPEEDY 1050 IST NICHT HAPPY-KOMPATIBEL!

Speedy 1050 N	198 — DM
Speedy 1050 T mit Trackanzeige und Summer	298 — DM
Nachrüstsatz Trackanz. u. Summer f. Speedy 1050 N	119 — DM
Anwender Handbuch Speedy 1050 mit ROM Listing	50 — DM
Einbau der Speedy 1050 in Ihr Laufwerk, zzgl.	60 — DM
Atari 1050 mit Speedy 1050 T	898 — DM

FORDERN SIE UNSEREN AUSFÜHRLICHEN PROSPEKT AN! HÄNDLER-ANFRAGEN ERWÜNSCHT!

Hardware – Software

Atari 800XL, mit 320 KRAM und 16-K-Bibomon	798 — DM
Atari 130XE, mit 192 KRAM und 16-K-Bibomon	849 — DM
Atari 1050 mit Speedy N	598 — DM
Atari 1050 mit Speedy T	898 — DM
Atari 1029 mit Grafikdruck Programm	498 — DM
Grafikdruck Programm einzeln	38 — DM
Mannesmann MT 80 anschlußfertig für XL XE	998 — DM
Okidata ML 182 anschlußfertig für XL XE	1098 — DM
Okidata ML 192, anschlußfertig für XL XE	1649 — DM
Printer Interface für Atari XL XE	148 — DM
Bit Burner Eprommer f. XL XE, b. 32 K EPROMs	248 — DM
SNAUT, NEU! Forth 83 Compiler auf Disk oder Kass.	98 — DM
Design Master das Super Programm für XL XE	19,80 DM
Soundmaschine, ebenfalls Super!	29,80 DM

SOLANGE VORRAT REICHT!

RGB-Monitor für XL XE mit leichten Gehäuseschäden (Kratzer) anschlußfertig für XL XE	698 — DM
ATARI POWER-Buch, viele Tips und Tricks	29 — DM
Die HEXENKÜCHE Buch m. sehr vielen Tips u. Tricks	29 — DM
Die HEXENKÜCHE, Disk z. Buch m. allen Progr.	19,80 DM

WIR SIND FACHHÄNDLER FÜR DAS NEUE SEGA VIDEOSPIEL SYSTEM

FORDERN SIE UNSERE AUSFÜHRLICHE PREISLISTE AN! KATALOG MIT CA. 170 SEITEN GEGEN 5,00 DM!

ALLE UNSERE PRODUKTE BEKOMMEN SIE IN ÖSTERREICH BEI DER FIRMA AMV IN WIEN!

kyan

Pascal Software Atari XL/XE

***kyan pascal* für die XL/XE Serie . . . DM 298,-**

kyan pascal ist ein mit DOS 2.5 arbeitender Compiler für die Atari XL/XE-Serie. Es umfaßt den vollen Jensen-Wirth Standard und eignet sich für den Anfänger genauso wie für den fortgeschrittenen Programmierer. Es ist blitzschnell und hat folgende Eigenschaften:

★ 6502 Maschinencode Compiler erlaubt das Einbinden von Assemblersource ★ Bildschirmeditor ★ Stringbefehle ★ Atari-Grafik und Soundunterstützung ★ Source Code Linking, Chaining und Random Files ★ 13stellige Floatingpoint Genauigkeit ★ Mit Tutorial/Referenzunterlagen ★

kyan pascal wird auf einer ungeschützten Single Density Diskette geliefert und benötigt nur 45 K Speicherplatz und ein Laufwerk.

kyan pascal für Ihren Atari Computer bei Ihrem Fachhändler oder direkt von uns

Händleranfragen

Witt Datentechnik, Am Denkmal 8
4802 Halle/Westfalen, 05201/4008

Compy-Shop, Gneisenaustr. 29, 4330 Mülheim/Ruhr
Telefon 0208-497169

PETER'S ASSEMBLERECKE

Software-Blitter

Ein neues Wort ist in aller Munde: der Blitter. Der Amiga hat ihn, der ST bekommt ihn, aber was hat es eigentlich damit auf sich?

Es handelt sich um einen hochspezialisierten Chip, der nur zum Kopieren von Speicherbereichen konstruiert wurde. Besonders geeignet erweist sich ein Blitter zum Bewegen von Objekten auf einem hochauflösenden Bildschirm, daher auch der Name, der für "Block Image Transferer" steht. In der Assemblerecke wollen wir Ihnen ein Programm vorstellen, das die Arbeit eines Blitters simuliert und gleichzeitig über zwei neue Basic-Befehle verfügt.

Im Gegensatz zur Player-Missile-Grafik oder Sprites, die über einen separaten Speicher für die Bilddaten verfügen, werden Bobs (Blitter Objects) direkt in den Bildspeicher kopiert. Dies hat den Vorteil, daß Objekte in fast beliebiger Größe und Anzahl bewegt werden können. Die einzige Einschränkung liegt in der vorhandenen Rechenzeit, denn je mehr bzw. je größere Bobs bewegt werden, desto größer ist auch die Datenmenge. Und diese Flut kann gewaltig sein. Während zur Bewegung eines Objektes mit PM-Grafik im günstigsten Fall die Veränderung eines einzigen Registers genügt, müssen bei Bobs häufig mehrere hundert oder tausend Bytes bewegt werden. Einem echten Blitter fällt das jedoch nicht schwer. Er kann per DMA sehr schnell auf den Speicher zugreifen, d.h. praktisch in jedem Arbeitstakt ein Byte lesen oder schreiben. Der 6502 benötigt dazu drei bis sieben Taktzyklen.

GET und PUT

Da unser Atari bekanntlich keinen Blitter hat, müssen wir uns mit einer Software-Lösung behelfen. Damit auch etwas praktisch Verwertbares dabei herauskommt, werden wir dem Atari Basic zwei neue Befehle (GET und PUT) hinzufügen. Sie kennen die beiden bereits? Irrtum, wer sich schon mit Basic-Dialekten für 16-Bit-Rechner befaßt hat, weiß, daß diese Befehle einen ganz neuen Sinn bekommen haben.

Mit GET kann man einem hochauflösenden Bildschirm einen rechteckigen Bereich entnehmen und seine Daten in einem String ablegen. PUT erledigt das Gegenteil, die Daten aus dem String werden in den HiRes-Screen zurückgeschrieben. Allerdings kann man den Bildausschnitt (nichts anderes als ein Bob) nun an jede beliebige Stelle des Bildschirms kopieren.

Diese beiden Befehle eröffnen dem Programmierer eine ganze Fülle neuer Möglichkeiten. Sie können nicht nur Objekte über den Bildschirm bewegen, sondern auch bei Windows behilflich sein und Symbole jeder Art anzeigen und wieder löschen.

Moderne Benutzeroberflächen wie das GEM des Atari ST (oder Intuition beim Amiga) arbeiten vorwiegend mit Symbolen (Icons) statt Texten. Wenn es Ihnen Spaß macht, können Sie mit GET und PUT ähnliche Programme schreiben. Sie benötigen dazu mehrere Symbole, die am Bildschirm gezeichnet und mit GET-Befehlen in Strings abgelegt werden. Durch PUT können diese Icons wieder sichtbar gemacht werden und so z.B. als Menü dienen. Auch Windows kann man mit unserem "Software-Blitter" verwalten. Bevor ein Window Teile des Bildes überschreibt, wird der Ausschnitt durch GET in einem String zwischengespeichert. Später läßt sich dieser per PUT an die alte Stelle zurückschreiben.

Zur Anwendung

Zunächst sollten Sie sich das Listing 1 näher ansehen. Es enthält im ersten Teil mehrere Demos mit Beispielen zu GET und PUT. Der zweite Teil (ab Zeile 30000) enthält das Maschinenprogramm des "Software-Blitter" als Data-Zeilen. Sollten Sie GET und PUT in eigenen Programmen einsetzen, brauchen Sie nur diesen Teil zu übernehmen, der das Maschinenprogramm in Page 6 ablegt. Die neuen Befehle können dann über USR angesprochen werden. Sinnvoll ist es auch, die Einsprungadressen des in Page 6 liegenden Programms wie im Beispiel als Variablen GGET=1536 (für Grafik GET) und GPUT=1539 zu definieren. Ein GET-Befehl wird dann so ausgeführt:

```
A=USR(GGET,X1,Y1,
X2,Y2,ADR(A$))
```

Die Koordinaten X1, Y1 geben die linke obere Ecke des Bildausschnitts an, X2 und Y2 die rechte untere. Als letzter Parameter muß die Adresse eines Strings eingesetzt werden (natürlich kann man statt A$ beliebige Namen einsetzen). Achten Sie nur darauf, daß in der DIM-Anweisung des Strings ausreichend Speicherplatz reserviert wurde. Den Platzbedarf können Sie wie folgt berechnen:

Bytes = (X2−X1+1) * (Y2−Y1+1)+2

Der PUT-Befehl wird ganz ähnlich aufgerufen (GPUT=1539):

```
A=USR(GPUT,X,Y,MOD,
ADR(A$))
```

Bei PUT müssen Sie nur die linke obere Ecke angeben, da Höhe und Breite des Ausschnitts im String bereits enthalten sind. Der Parameter MOD bestimmt, wie PUT den Ausschnitt einsetzt. Dies geschieht durch folgende Codes:

0 OR-Modus: Der Ausschnitt wird dem bereits vorhandenen Bild durch eine Oder-Verknüpfung überlagert, z.B. ein Bob mit dem Hintergrund.

64 EXOR-Modus: Der Ausschnitt wird Exklusiv-Oder mit dem vorhandenen Bild verknüpft. Schreibt man den Ausschnitt ein zweites Mal an die gleiche Stelle, so verschwindet er wieder. Bobs können damit zerstörungsfrei über den Hintergrund bewegt werden.

128 PSET-Modus: Hier wird der Bildteil vom Ausschnitt einfach überschrieben. Diese Methode eignet sich zum Zurückkopieren von Hintergrundinformationen (s. Beispiel).

Noch ein Wort zu den Koordinaten: Der "Software-Blitter" ist auf GRAPHICS 8 zugeschnitten. Y kann daher Werte von 0 bis 191 annehmen, während für X anstatt 0 bis 319 nur der Bereich von 0 bis 39 zulässig ist. Diese Einschränkung ergibt sich aus dem Aufbau des Video-Speichers bei hochauflösender Grafik. Je acht Bildpunkte sind in einem Byte zusammengefaßt und dies ist die kleinste Einheit, die unser "Blitter" verarbeiten kann. Die gewohnten Koordinaten von 0 bis 319 müssen daher für GET und PUT jeweils durch acht geteilt werden.

Man könnte den "Software-Blitter" natürlich auch so auslegen, daß er die Byte-Grenzen ignoriert und alle 320 möglichen X-Koordinaten zuläßt. Tatsächlich war die erste Version des Maschinenprogramms so angelegt, aber das Ergebnis war fatal: Das Programm erreichte nur noch ein Schneckentempo, da nun zusätzlich die Bit-Positionen innerhalb der Bytes verschoben und eine Menge von Randbedingungen beachtet werden mußten. Es war für jegliche Bewegungseffekte viel zu langsam.

Hier hat nun ein Hardware-Blitter die Nase weit vorne, da er über sogenannte Barrel Shifters verfügt. Stellen Sie sich dazu bitte folgendes vor: Wenn ein Byte um vier Bitstellen verschoben werden soll, muß der Prozessor vier Schiebebefehle (z.B. ROL) nacheinander ausführen. Ein Barrel-Shifter liefert das gleiche Ergebnis in einem einzigen Arbeitstakt, eine Zeitspanne, in der der Prozessor nicht einmal einen einzigen Befehl ausführen könnte.

Trotz dieser Einschränkungen ist der "Soft-Blitter" sehr nützlich. Wenn Sie Listing 1 eingetippt und die Demos angeschaut haben, werden Ihnen bestimmt eine Menge neuer Einsatzgebiete enthalten.

Bewegung

Im Demo wird auch Bewegung mit dem "Soft-Blitter" erzeugt. Das Ergebnis ist zwar nicht ganz so ruckfrei und flimmerfrei, wie man es sich wünschen würde, aber immerhin wird ein 40 mal 30 Punkte großes Objekt (ein Flugzeug) zerstörungsfrei über den Hintergrund bewegt – und das in Basic!

Zu Beginn jedes Bewegungsschritts wird ein Ausschnitt des Hintergrunds in den String HG$ gerettet. Anschließend wird das Flugzeug aus GR$ mit einem PUT-Befehl (im Oder-Modus) dem Bild an dieser Stelle überlagert. Nun beginnt eine kleine Warteschleife, die dafür sorgt, daß das Flugzeug eine Weile sichtbar ist. Der in HG$ gespeicherte Hintergrund überschreibt dann mit einem erneuten PUT-Befehl (diesmal im PSET-Modus) das Bild mit dem zuvor geretteten Hintergrund. Damit ist der ursprüngliche Zustand des Bildes wiederhergestellt. Im nächsten Schleifendurchlauf kann das Flugzeug ein Stück entfernt neu gezeichnet werden. Da die Zeitspanne vom Löschen bis zum erneuten Erscheinen dank dem "Soft-Blitter" sehr kurz ist, entsteht der Eindruck einer Bewegung.

Wesentlich einfacher wäre die Bewegung mit dem EXOR-Modus zu erzielen, da hier der Buffer für den Hintergrund entfällt. Allerdings wird bei dieser Methode das Objekt ziemlich unansehnlich, wenn der Hintergrund dicht beschrieben ist. Probieren Sie es aber ruhig mal aus!

Auch das lästige Flackern des Objekts könnte man beseitigen. Man müßte nur einen zweiten GRAPHICS-8-Bildschirm anlegen und dann per Page Flipping zwischen beiden hin- und herschalten. Während der "Blitter" an einem Bild arbeitet, wird das andere angezeigt, und sobald ein neues fertiggestellt ist, werden die Rollen getauscht. Diese Technik ist sehr wirkungsvoll und wäre vielleicht ein Thema für eine zukünftige Assemblerecke.

Blitter intern

Wie immer werfen wir auch diesmal einen Blick hinter die Kulissen des Maschinenprogramms (Listing 2). Nach einigen Definitionen von Variablen, die zur Beschleunigung alle in die Zero Page gelegt wurden, findet sich eine kleine Sprungtabelle zu den Routinen GET und PUT.

Die GET-Routine beginnt ihre Arbeit, indem sie alle Parameter vom Stack nimmt und in die zugehörigen Variablen einträgt. Danach wird die Größe des Ausschnitts berechnet und gleich geprüft, ob die Koordinaten richtig eingegeben wurden, d.h. der linke Rand nicht größer als der rechte ist. Die berechneten Größen werden in den String als erstes und zweites Byte eingetragen, damit die PUT-Routine später weiß, wie groß der Ausschnitt ist.

Im Unterprogramm BERZEIG (berechne Zeiger) wird die Anfangsadresse der Zeile im Video-Speicher berechnet, wo der Ausschnitt beginnt. Mit Hilfe dieses Zeigers kann die folgende Schleife (ab Label GETZEIL) Byte für Byte aus dem Video-Speicher entnommen und per Unterprogramm PUTSTR in den gewünschten String eingetragen werden. Entscheidend ist, daß man die Bytes aus dem Video-Speicher nicht kontinuierlich entnehmen darf. Sehen Sie sich dazu Bild 1 an, das einen Teil des Video-Speichers darstellen soll. Beginnend am linken Rand des gewählten Ausschnitts müssen soviel Bytes entnommen werden, bis der rechte Rand angetroffen wird. Nun muß ein Sprung zum linken Rand der nächsten Zeile erfolgen. Die Größe der Lücke hängt somit von der Breite des Ausschnitts und der Anzahl der Bytes pro Zeile ab. Man nennt diese Lücke gewöhnlich das Modulo.

Diese Situation wäre eine echte Anwendung für einen Hardware-Blitter. Ein solcher Chip hat Register für die Breite des Ausschnitts und das Modulo, und zwar sowohl für die Quelle der Daten (in unserem Fall das Video-RAM) als auch für das Ziel (hier der Basic-String). Da die Daten im String kontinuierlich abgelegt werden, wäre im vorliegenden Fall das Modulo eben null. Sie sehen, der Name "Soft-Blitter" ist nicht aus dem Ärmel geschüttelt: die vorliegende Routine imitiert tatsächlich die Funktion eines Hardware-Blitters, wenn auch in stark vereinfachter Form.

In ihrem Aufbau entspricht die PUT-Routine der gerade besprochenen GET-Funktion, nur ist die Richtung des Datenflusses umgekehrt. Die Quelle (String) besitzt nun einen Modulo von null, das Ziel (Video-RAM) einen Modulo von Zeilenlänge minus Breite des Ausschnitts. Die Modulo-Werte sind in beiden Routinen fest einprogrammiert und können daher nicht frei gewählt werden. Durch diese Maßnahme wird das Programm kürzer und schneller.

Computer-Kontakt das Heft mit den preisgünstigen Kleinanzeigen

Interessant ist vielleicht noch die Abfrage des Modus durch den BIT-Befehl. Dies ist eine kritische Stelle im Programm. Je länger die Abfrage dauert, desto langsamer wird das Programm, da sie für jedes bearbeitete Byte erfolgen muß. Ein Befehl mehr oder weniger kann sich dabei ganz hübsch auswirken. Die beste Lösung wäre sicherlich, für jeden Modus eine eigene "Blitter"-Routine zu schreiben. Nur leider wird dann das Programm für Page 6 viel zu lang.

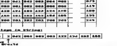
Ein wenig Aufmerksamkeit verdient auch die Berechnung der Zeilenadresse im Unterprogramm BERZEIG. Es handelt sich nicht um eine allgemeine Routine zur Multiplikation. Vielmehr werden hier trickreich Bit-Verschiebungen sowie Additionen eingesetzt, um eine Multiplikation mit 40 zu erreichen. Aufmerksame Leser der Assemblerecke kennen dieses Prinzip bereits aus früheren Folgen. Warum nun gerade mal 40? Ganz einfach: das ist die Anzahl der Bytes pro Zeile in GRAPHICS 8.

Andere Grafikstufen

Obwohl der "Soft-Blitter" nur für den Einsatz mit GRAPHICS 8 konzipiert ist, kann man ihn auch mit einigen anderen Grafikstufen verwenden. Denkbar wären z.B. GRAPHICS 15 oder auch GRAPHICS 0. Im letzteren Fall könnte man recht effektiv mit Windows arbeiten, denn dem "Blitter" ist es egal, ob die bearbeiteten Daten aus einem Text- oder HiRes-Bildschirm stammen. Aufpassen muß man jedoch mit den Y-Koordinaten, die dann nur noch von 0 bis 23 zulässig sind (es erfolgt keine Prüfung!); außerdem fänden die Modi OR bzw. EXOR des PUT-Befehls keine vernünftige Verwendung mehr. Ähnliche Einschränkungen ergeben sich mit den Farbgrafik-Stufen wie 7 oder 15. Da nun zwei nebeneinander liegende Bits die Farbinformation bestimmen, kommt es beim OR- bzw. EXOR-Modus zur Verfälschung der Farben. GET und PUT im PSET-Modus bleiben jedoch durchaus brauchbar. Es lohnt sich bestimmt, damit zu experimentieren.

Peter Finzel

Aufbau des Video-RAMs



MC-Generator

```
100 REM ******************************
110 REM BLITTER-OBJECTS-(BOBS)-DEMO
120 REM
130 REM P. FINZEL               1986
140 REM ******************************
150 REM
200 DIM BR$(300),HG$(300)
210 BGET=1536:BPUT=1539
220 GRAPHICS 2+16
230 POSITION 2,4:PRINT #6;"SOFT-BLITTE
R DEMO"
240 POSITION 1,6:PRINT #6;"EINEN MOMEN
T BITTE"
250 GOSUB 30000
300 REM * Zeichne Flugzeug
305 GRAPHICS 8:COLOR 1
310 PLOT 55,20:DRAWTO 63,0:POSITION 47
,12:GOSUB 700
320 DRAWTO 15,12:POSITION 0,20:GOSUB 7
00
330 PLOT 43,30:DRAWTO 38,20:PLOT 25,20
:POSITION 35,30:GOSUB 700
335 REM * Bild im String
340 X=USR(BGET,0,0,8,30,ADR(BR$))
390 REM * viele Flugzeuge...
400 GRAPHICS 8+16:SETCOLOR 2,1,0
410 FOR X=0 TO 32 STEP 8
420 FOR Y=0 TO 140 STEP 35
430 A=USR(BPUT,X,Y,0,ADR(BR$))
440 NEXT Y:NEXT X
450 FOR Y=0 TO 140 STEP 35
460 FOR X=0 TO 32 STEP 8:SOUND 0,Y,10,
10
470 A=USR(BPUT,X,Y,64,ADR(BR$))
480 SOUND 0,0,0,0:FOR I=0 TO 7:GOSUB 8
00:NEXT I
490 NEXT X:NEXT Y
500 REM * BEWEGUNG
510 M=69:GOSUB 900:SOUND 0,15,0,10
515 Y=100:FOR X=32 TO 0 STEP -1:Y=Y-2
520 A=USR(BGET,X,Y,X+7,Y+30,ADR(HG$))
530 A=USR(BPUT,X,Y,0,ADR(BR$))
540 FOR I=0 TO 4:GOSUB 800:NEXT I
560 A=USR(BPUT,X,Y,128,ADR(HG$))
570 NEXT X:SOUND 0,0,0,0
590 GOTO 400
700 REM * XIO-FILLROUTINE
710 POKE 765,1:XIO 18,#6,0,0,"S:"
790 RETURN
800 REM * 1/50 SEC. WARTEN
810 T=PEEK(20)
820 IF PEEK(20)=T THEN 820
830 RETURN
900 REM * BILD AUFBAUEN
910 GRAPHICS 8:SETCOLOR 2,8,0
920 FOR I=0 TO 319 STEP 31:PLOT 159,M:
DRAWTO I,M+90:NEXT I
930 I=0
940 Q=I*I+M:IF Q<=159 THEN PLOT 0,Q:DR
AWTO 319,Q:I=I+1:GOTO 940
990 RETURN
30000 REM * Soft-Blitter einrichten
30010 S=0:RESTORE 30100
30020 FOR A=1536 TO 1785:READ D:POKE A
,D:S=S+D:NEXT A
30030 IF S<>34981 THEN ? "DATEN-FEHLER
!":STOP
30090 RETURN
30100 DATA 76,6,6,76,98,6,216,104,104,
104,133,218,104,104,133,217
30110 DATA 104,104,133,220,104,104,133
,219,104,133,215,104,133,214
30120 DATA 56,165,220,229,218,176,1,96
,133,221,230,221,96,165,219
30130 DATA 229,217,176,1,96,24,105,1,1
33,219,32,239,6,165,221,32,239
30140 DATA 6,32,175,6,164,218,165,221,
133,216,177,212,32,239,6,200
30150 DATA 198,216,208,246,24,165,212,
105,40,133,212,144,2,230,213
30160 DATA 198,219,208,225,96,216,104,
104,104,133,218,104,104,133
30170 DATA 217,104,104,133,222,104,133
,215,104,133,214,32,228,6,133
30180 DATA 219,32,228,6,133,221,32,175
,6,165,221,133,216,164,218,32
30190 DATA 228,6,36,222,48,8,112,4,17,
212,80,2,81,212,145,212,200
30200 DATA 198,216,208,234,24,165,212,
105,40,133,212,144,2,230,213
30210 DATA 198,219,208,213,96,165,217,
133,212,169,0,133,213,6,212
30220 DATA 38,213,6,212,38,213,24,165,
212,101,217,133,212,144,2,230
30230 DATA 213,6,212,38,213,6,212,38,2
13,6,212,38,213,24,165,212,101
30240 DATA 88,133,212,165,213,101,89,1
33,213,96,162,0,161,214,230
30250 DATA 214,208,2,230,215,96,162,0,
129,214,230,214,208,2,230,215,96
```

Assemblerlisting

```
**************************************
*             SOFTWARE-BLITTER    V1.02
*
*         PUT/GET fuer Graphik
*          (Byte-Aufloesung)
*
*  Peter Finzel                    1986
**************************************
*
ZLAENGE  EQU 40  Zeilenlaenge in BYTE
*
SAVMSC   EQU $58 Zeiger auf Video-RAM
*
* Variable in der Zero-Page
*
```

```
ZEIGER   EQU $D4 Pointer im Screen
STRING   EQU $D6 Pointer im String
BZAEHL   EQU $D8 Zaehler
ORAND    EQU $D9 Koordinaten: oberer Rand
LRAND    EQU $DA linker Rand
URAND    EQU $DB unterer Rand
RRAND    EQU $DC rechter Rand
BLAENGE  EQU $DD Laenge fuer Blitter
MODUS    EQU $DE Modus fuer PUT 0: Oder
*                64:Exor      128: Pset
*
         ORG $0600
*
* Sprungleiste
*
         JMP GET
         JMP PUT
*
* GET-Routine fuer Bildausschnitte
*  Aufruf:
*  A=USR(1536,X1,Y1,X2,Y2,ADR(A$))
*
GET      CLD
         PLA          Anz. der Parameter
         PLA          Linker Rand MSB
         PLA          L. Rand LSB
         STA LRAND
         PLA          Oberer Rand MSB
         PLA          Oberer Rand LSB
         STA ORAND
         PLA          rechter Rand MSB
         PLA          und LSB
         STA RRAND
         PLA          unterer Rand MSB
         PLA          und LSB
         STA URAND
         PLA          Adresse des Strings
         STA STRING+1
         PLA
         STA STRING

         SEC          Anz. der horizontalen
         LDA RRAND    Bytes ermitteln
         SBC LRAND
         BCS LENOK    ueberhaupt moeglich?
         RTS          Linker > Rechter!

LENOK    STA BLAENGE Laenge merken
         INC BLAENGE Laenge= R-L+1

         SEC           Anzahl der Zeilen
         LDA URAND     ermitteln
         SBC ORAND
         BCS HOENOK    moeglich?
         RTS           Oben > Unten!

HOENOK   CLC           Zeilenanzahl
         ADC #1        korrigieren
         STA URAND     und in URAND eintragen
         JSR PUTSTR    Hoehe & Breite des
         LDA BLAENGE   Ausschnittes im
         JSR PUTSTR    String eintragen

MSS1     JSR BERZEIG Zeiger in Video-Ram
*
```

```
* Soft-Blitter fuer GET
*
GETZEIL  LDY LRAND        linker Rand
         LDA BLAENGE      Byte-Zaehler
         STA BZAEHL       einrichten
GETBYTE  LDA (ZEIGER),Y Wert aus Video/RAM
         JSR PUTSTR       in String ablegen
         INY              naechstes Byte
         DEC BZAEHL       schon ganze Zeile
         BNE GETBYTE      nein ->
         CLC              Zeiger auf naechste
         LDA ZEIGER       Zeile im Video-RAM
         ADC #ZLAENGE     richten (plus 40)
         STA ZEIGER
         BCC GET2         jetzt MSB
         INC ZEIGER+1
*
GET2     DEC URAND        schon alle Zeilen?
         BNE GETZEIL      nein -->
         RTS
*
* PUT-ROUTINE fuer Bildausschnitte
*  Aufruf:
*  A=USR(1539,X,Y,Mod,Adr(A$))
*
PUT      CLD
         PLA           Anz. der Argumente
         PLA           linker Rand MSB
         PLA           und LSB
         STA LRAND
         PLA           Oberer Rand MSB
         PLA           und LSB
         STA ORAND
         PLA           Modus-Byte
         PLA           nur LSB!
         STA MODUS
         PLA           Stringadresse
         STA STRING+1
         PLA
         STA STRING
*
         JSR GETSTR  Anzahl d. Zeilen
         STA URAND   des Ausschnittes
         JSR GETSTR  Hor. Laenge des
         STA BLAENGE Ausschnittes
*
EGHZ2    JSR BERZEIG Zeiger in Video-RAM
*
* Soft-Blitter fuer PUT
*
PUTZEIL  LDA BLAENGE
         STA BZAEHL       Zaehler Zeilenlaenge
         LDY LRAND        linker Rand

PUTBYTE  JSR GETSTR       Zeichen aus String
         BIT MODUS        welcher Modus?
         BMI PSET         ->  PSET
         BVS EXOR         -> Ex-Oder
         ORA (ZEIGER),Y   Oder
         BVC PSET         verzweigt immer
EXOR     EOR (ZEIGER),Y Exor-Verkn.
PSET     STA (ZEIGER),Y in Video-RAM
         INY              naechst. Spalte
         DEC BZAEHL       Zeile fertig?
         BNE PUTBYTE      nein -->
         CLC                naechste Zeile
         LDA ZEIGER         Zeiger=Zeiger+40
         ADC #ZLAENGE
         STA ZEIGER
         BCC PUT2
         INC ZEIGER+1       MSB
*
PUT2     DEC URAND          alle Zeilen fertig?
         BNE PUTZEIL        nein -->
         RTS
*
* Zeiger auf erstes Byte berechnen
*            ZEIGER = ORAND * 40
*
BERZEIG  LDA ORAND          LSB Zeiger=ORAND
         STA ZEIGER
         LDA #0             MSB Zeiger=0
         STA ZEIGER+1
         ASL ZEIGER         Zeiger=Zeiger*2
         ROL ZEIGER+1
         ASL ZEIGER         Zeiger=Zeiger*2
         ROL ZEIGER+1       ergibt ORAND*4
         CLC
         LDA ZEIGER         Zeiger=Zeiger+ORAND
         ADC ORAND          bisher berechnet:
         STA ZEIGER         Zeiger=ORAND*5
         BCC BCRADR1
         INC ZEIGER+1
BCRADR1  ASL ZEIGER         Zeiger=Zeiger*2
         ROL ZEIGER+1
         ASL ZEIGER         Zeiger=Zeiger*2
         ROL ZEIGER+1
         ASL ZEIGER         Zeiger=Zeiger*2
         ROL ZEIGER+1       damit:
         CLC                ZEIGER=ORAND*40
         LDA ZEIGER         nun noch die
         ADC SAVMSC         Basisadresse des
         STA ZEIGER         Video-RAMs addieren
         LDA ZEIGER+1
         ADC SAVMSC+1
         STA ZEIGER+1
         RTS
*
* Routinen zur String-Verwaltung
*
GETSTR   LDX #0             Ein Byte aus
         LDA (STRING,X)     dem String holen
         INC STRING         und Stringzeiger
         BNE GS1            weiterschalten
         INC STRING+1
GS1      RTS
*
PUTSTR   LDX #0             Ein Zeichen in
         STA (STRING,X)     String eintragen
         INC STRING         und Stringzeiger
         BNE PS1            weiterzaehlen
         INC STRING+1
PS1      RTS
```

Wombel auf dem Eis

"Wombel", ein Spiel für Atari 800 XL, ist mehr als nur ein Reaktions- und Geschicklichkeitsspiel. Es verlangt durchdachtes Handeln, ohne daß Action und Spielwitz zu kurz kommen. Das Besondere an "Wombel" ist nämlich, daß Steine und blaue Eisblöcke von allein in ein anliegendes Feld hüpfen, was für den kleinen, weißen Pinguin Wombel zwar nicht sehr gefährlich ist, aber seine ganzen Pläne über den Haufen werfen kann. So bleibt "Wombel" immer spannend und spielenswert.

Wichtigstes Ziel ist es, drei blinkende Eisblöcke in eine Reihe zu schieben. Dazu muß man sich erst einmal eine geeignete Stelle des Spielfelds aussuchen, an die alle drei Blöcke geschoben werden können, ohne daß ein Hindernis (Stein, Schneewesen) dazwischen ist. Das ist gar nicht so einfach, meistens springt auch noch ein Stein oder Eisblock in den Weg, und man muß umdenken. Besonders schlimm ist es, wenn ein Stein so hüpft, daß ein blinkender Eisblock nicht mehr verschoben werden kann. Was nun? Hierfür gibt es pro Runde eine blinkende Krafttablette, die für solche Fälle aufgehoben werden sollte. Man kann mit ihr aber natürlich auch einen Stein beseitigen, der im Weg liegt. Nimmt Wombel nämlich diese Tablette, so hat der Spieler kurze Zeit die Möglichkeit, Steine aufzulösen. Dazu fährt er auf einen Stein zu, bleibt direkt vor ihm stehen und drückt den Feuerknopf am Joystick. Berühren darf man aber während dieser Zeit die Steine nicht.

In großer Zahl gibt es auf Wombels Spielfeld blaue Eisblöcke, mit denen sich eine Art Wand bilden läßt, an die man die blinkenden Eisblöcke schiebt. Natürlich kann man die drei blinkenden Blöcke auch an der Spielfeldumrandung aufreihen, doch der Weg dorthin ist leider meistens versperrt. Außerdem läßt sich ein Eisblock, der am Spielfeldrand liegt, in die Gegenrichtung nicht mehr verschieben. Die Gefahr ist also groß, daß er eingeschlossen wird.

Zurück zu den zahlreichen blauen Eisblöcken: Wombel muß sie sehr oft aus dem Weg räumen oder gar auflösen, wenn einer nicht mehr verschiebbar ist. Außerdem kann man mit allen Eisblöcken Schneewesen überfahren, was aber nur von links oder rechts geht, damit es nicht zu einfach wird. Passen Sie aber immer auf, wo sich die zwei Eisstrahlwesen befinden. Sie sind immer übereinander, eines am oberen, eines am unteren Spielfeldrand. Sie tauchen an irgendeiner Position auf und bilden nach kurzer Zeit zwischen sich einen senkrechten Eisstrahl, den Wombel nicht berühren darf. Tauchen die Eisstrahlwesen also plötzlich direkt über und unter Wombel auf, sollte man sofort nach links oder rechts ausweichen, sonst verliert Wombel eines seiner drei Bildschirmleben. Von Runde zu Runde wird die Reaktionszeit kürzer, genauso wie von Runde zu Runde Steine und Eisblöcke öfter springen und immer mehr Schneewesen die Unwesen treiben. Fangen Sie ruhig einmal mit Runde 9 an (mit SELECT zu Beginn wählbar), dann bekommen Sie gleich 9 Schneewesen auf einmal zu sehen. In Runde 10 sind es dann 10 Schneewesen gleichzeitig, das heißt, in jeder Zeile des Spielfelds eines! Sind Sie gut genug, können Sie sich am Ende des Spiels in die Highscore-Tabelle eintragen, die zu Beginn jedes Spiels wieder gezeigt wird.

Jörg Löffler

ATARI 1029
* HARDCOPY *

- Ausdruck in allen Größen von der Briefmarke bis zur vollen DIN A4 Seite
- Ausdruck invertiert
- Ausdruck gespiegelt
- Ausdruck inv. und gesp.
- Ausschnittvergrößerung
- Ausdruck und Speichern des Ausschnitts

Demos in Originalgröße kommentiert. Diskette VOLLE 64 KByte
40,– DM zzgl. Versand Bestellung
[illegible]

Jürgen Dörr
Einsteinstraße 6
6520 Worms 26
Tel. 06241/34140

```
0 REM
1 REM        M U N D E L
2 REM
3 REM [illegible]
4 REM
5 REM   ATARI 800XL + Joystick
6 REM
7 REM
10 GOSUB 12000
11 GOSUB 7000:L=0
15 REM [illegible]
20 P1=PEEK(63252):P2=PEEK(63283)
30 XP=PEEK(203):YP=PEEK(204):SOUND 1,L,L,L
40 IF P1=1 OR P2=1 THEN POKE 2720,150:CL=37:FA=216,GOTO 200+PEEK(205)*100
50 IF P1=8 OR P2=8 THEN POKE 2720,16:CL=215:FA=120:GOTO 200+PEEK(105)*100
60 IF P1=2 OR P2=2 OR P1=4 OR P2=4 OR PEEK(53283)>L THEN 2000
70 IF SGP(L)>VL THEN GOSUB 200
75 IF ZL>L THEN ZL=ZL-L:GOTO 100
77 COLOR 228:PLOT 39,2:PLOT X5,13:X5=INT(RND(L)*18)+3:COLOR 171:PLOT 39,2:PLOT X5,13:ZL=VZ
80 POKE 87,L:POSITION 20,L:? #6;SD-MEN(19);" ":POKE 87,1:IF PEEK(19)=06 THEN 2000
85 IF ER>L THEN ER=ER-1:GOSUB 800:IF ER<=L THEN POKE 710,124:POKE 1701,104
90 SOUND 2,L,L,L:POKE 53278,L:POKE 77,L:GOTO 20
95 REM [illegible]
100 IF ZL>L THEN 80
110 RE=INT(RND(L)*500+61):POKE 53282,RE:SOUND 1,60,12,8
120 IF RE>PEEK(200) AND RE<PEEK(207)+8 THEN 2000
130 POKE 53252,L:GOTO 80
240 REM [illegible]
250 XX=INT(RND(L)*18)+1:YR=INT(RND(L)*18)+3
260 LOCATE XX,YR,CX:IF CX<>130 AND CX<>32 THEN RETURN
270 XX=INT(RND(L)*3)-1:YY=INT(RND(L)*3)-1
275 LOCATE XX+XX,YR+YY,C:IF C<>32 AND C<>159 THEN RETURN
280 SOUND 2,121,12,8:COLOR 32:PLOT XX,YR
280 XX=XR+XX:YR=YR+YY:COLOR 162:PLOT XX,YR:SOUND 2,255,12,0:FOR T=1 TO 2
290 IF PEEK(53282)=4 OR PEEK(63251)=6 THEN 2000
293 NEXT T
295 COLOR CX:PLOT XX,YR:SOUND 2,L,L,L:RETURN
297 REM [illegible]
300 X=(XP-48)/8+0.8:Y=YP/16-2:XM=X*8+48:YM=Y*16+17
305 IF CL=37 THEN LOCATE X,Y,CX:IF CX=84 THEN 970
310 LOCATE X+2,Y,CX:IF CX<>32 THEN 720
320 POKE 708,FA:POKE 207,YM:POKE 206,XM:COLOR 32:PLOT X,Y:SM=20:SM=2.5:POKE 53278,L
330 SC=SC+10:FOR I=XM TO XM+7:SD=SD-SM:SOUND 2,50,10,10:POKE 206,I
335 IF PEEK(53282)=4 THEN IF RR=L THEN RR=PEEK(10912+Y-3):GOSUB 500
340 NEXT I:SM=L:X=X+1:XM=XM+8:LOCATE X+1,Y,CX:IF CX=32 THEN 330
350 COLOR CL:PLOT X,Y:POKE 206,L:POKE 207,L:FOR I=15 TO 1 STEP -0.4:SOUND 2,200,12,I:NEXT I:SOUND 2,L,L,L
360 POKE 708,L:GOSUB 750:IF CL<>37 THEN 50
370 TRAP 90:LOCATE X+1,Y,CX:IF CX=17 THEN LOCATE X+2,Y,CX:IF CX=37 THEN 4700
375 LOCATE X,Y+1,CX:IF CX=37 THEN GOSUB 750
380 LOCATE X,Y-1,CX:IF CX=37 THEN GOSUB 700
390 GOTO 50
395 REM [illegible]
400 X=(XP-48)/8-1:Y=YP/16-1:XM=X*8+48:YM=Y*16+17
405 IF CL=37 THEN LOCATE X,Y,CX:IF CX=84 THEN 970
410 TRAP 700:LOCATE X-1,Y,CX:IF CX<>32 THEN 710
420 POKE 708,FA:POKE 207,YM:POKE 206,XM:COLOR 32:PLOT X,Y:SD=26:SM=1:POKE 53278,L
430 SC=SC+10:FOR I=XM TO XM-7 STEP -1:SD=SD-SM:SOUND 2,50,10,10:POKE 206,I
435 IF PEEK(53282)=4 THEN IF RR=L THEN RR=PEEK(10912+Y-3):GOSUB 500
440 NEXT I:SM=L:X=X-1:XM=XM-8:TRAP 460:LOCATE X-1,Y,CX:IF CX=32 THEN 430
450 COLOR CL:PLOT X,Y:POKE 206,L:POKE 207,L:FOR I=15 TO 1 STEP -0.4:SOUND 2,200,12,I:NEXT I:SOUND 2,L,L,L
460 POKE 708,L:GOSUB 750:IF CL<>37 THEN 50
470 TRAP 50:LOCATE X-1,Y,CX:IF CX=17 THEN LOCATE X-2,Y,CX:IF CX=37 THEN 4700
475 LOCATE X,Y+1,CX:IF CX=37 THEN GOSUB 700
480 LOCATE X,Y-1,CX:IF CX=37 THEN GOSUB 700
490 GOTO 50
495 REM [illegible]
500 X=(XP-48)/8:Y=YP/16-1:XM=X*8+48:YM=Y*16+4
505 IF CL=37 THEN LOCATE X,Y,CX:IF CX=84 THEN 970
510 LOCATE X,Y-1,CX:IF CX<>32 THEN 710
520 POKE 708,FA:POKE 207,YM:POKE 206,XM:COLOR 32:PLOT X,Y:SD=20:SM=3.5:POKE 53278,L
530 SC=SC+10:FOR I=YM TO YM-14 STEP -2:SD=SD-SM:SOUND 2,50,10,20:POKE 207,I
535 P2=PEEK(53253):IF P2=1 OR P2=7 THEN 2000
540 NEXT I:SM=L:Y=Y-1:YM=YM-16:LOCATE X,Y-1,CX:IF CX=32 THEN 530
560 COLOR CL:PLOT X,Y:POKE 206,L:POKE 207,L:FOR I=15 TO 1 STEP -0.4:SOUND 2,200,12,I:NEXT I:SOUND 2,L,L,L
565 POKE 708,L:GOSUB 750:IF CL<>37 THEN 50
570 TRAP 50:LOCATE X,Y-1,CX:IF CX=37 THEN LOCATE X,Y-2,CX:IF CX=37 THEN 4700
575 LOCATE X+1,Y,CX:IF CX=37 THEN GOSUB 770
580 LOCATE X-1,Y,CX:IF CX=37 THEN GOSUB 780
590 GOTO 50
595 REM [illegible]
600 X=(XP-48)/8:Y=YP/16:XM=X*8+48:YM=Y*16+14
605 IF CL=17 THEN LOCATE X,Y,CX:IF CX=84 THEN 970
610 LOCATE X,Y+1,CX:IF CX<>32 THEN 710
620 POKE 708,FA:POKE 207,YM:POKE 206,XM:COLOR 32:PLOT X,Y:SD=32:SM=4:POKE 53278,L
630 SC=SC+10:FOR I=YM TO YM+14 STEP 2:SD=SD-SM:SOUND 2,50,10,10:POKE 207,I
635 P2=PEEK(53253):IF P2=1 OR P2=7 THEN 2000
640 NEXT I:SM=L:Y=Y+L:YM=YM+16:LOCATE X,Y+L,CX:IF CX=32 THEN 630
650 COLOR CL:PLOT X,Y:POKE 206,L:POKE 207,L:FOR I=15 TO 1 STEP -0.4:SOUND 2,200,12,I:NEXT I:SOUND 2,L,L,L
660 POKE 708,L:GOSUB 750:IF CL<>37 THEN 50
670 TRAP 50:LOCATE X,Y+1,CX:IF CX=37 THEN LOCATE X,Y+2,CX:IF CX=17 THEN 4700
675 LOCATE X+1,Y,CX:IF CX=37 THEN GOSUB 770
680 LOCATE X-1,Y,CX:IF CX=17 THEN GOSUB 780
690 GOTO 50
700 REM [illegible]
710 IF CL=37 THEN 2000
720 FOR I=15 TO 9 STEP -1:FOR II=107 TO 190
725 IF PEEK(53284)=1 THEN 2000
730 SOUND 2,II,12,I:COLOR II:PLOT X,Y:NEXT II:NEXT I:SOUND 2,L,L,L
```

```
740 COLOR 32:PLOT X,Y:GOTO 20
750 LOCATE X,Y+2,CX:IF CX=37 THEN POP
:GOTO 4720
755 LOCATE X,Y-1,CX:IF CX=37 THEN POP
:GOTO 4700
757 RETURN
760 LOCATE X,Y-2,CX:IF CX=37 THEN POP
:GOTO 4720
765 LOCATE X,Y+1,CX:IF CX=37 THEN POP
:GOTO 4720
767 RETURN
770 LOCATE X+2,Y,CX:IF CX=37 THEN POP
:GOTO 4720
775 LOCATE X-1,Y,CX:IF CX=37 THEN POP
:GOTO 4700
777 RETURN
780 LOCATE X-2,Y,CX:IF CX=37 THEN POP
:GOTO 4700
785 LOCATE X+1,Y,CX:IF CX=37 THEN POP
:GOTO 4700
790 POSITION 7,L:? NE,SC:RETURN
795 REM [illegible]
800 IF STRIG(1) THEN RETURN
810 SOUND 1,L,L,L:POKE 1546,10:FOR X=2
24 TO L STEP -20:SOUND 3,1,10,10
815 NEXT X:SOUND 1,L,L,L:KR=KR-L:IF KR
<L THEN POKE 710,124:POKE 1761,184
820 GOTO PEEK(209)+829
830 X=(XP-48)/8+1:Y=YP/16-1:GOTO 840
831 X=(XP-48)/8-1:Y=YP/16-1:GOTO 840
832 X=(XP-48)/8:Y=YP/16-2:GOTO 840
833 X=(XP-48)/8:Y=YP/16+0.5:GOTO 840
840 LOCATE X,Y,G:IF G<>2 THEN POKE 154
6,200:RETURN
850 FOR I=68 TO L STEP -1:SOUND 2,I*30
,10,10-I/2:POKE 53768,I:COLOR 10+INT(RND(
L)*4)+27:PLOT X,Y:NEXT I
860 SOUND 1,L,L,L:SC=SC+50:GOSUB 790
870 COLOR 32:PLOT X,Y:POKE 1546,200:RE
TURN
895 REM [illegible] KAPUTT
900 POKE 1546,10:SOUND 2,L,L,L:SOUND 3
,L,L,L:SC=SC+300:GOSUB 790
910 FOR YY=16 TO L STEP -1:FOR XX=70 T
O 100 STEP 10
920 POKE 38512+Y-1,XX:SOUND 2,XX*2,10,
11:NEXT XX:POKE 38512+Y-1,1:NEXT YY
930 POKE 38512+Y-3,L:POKE 1546,224:RET
URN
950 REM [illegible] GENOMMEN
970 POKE 1761,165:POKE 39150,189:FOR I
=L TO 30:SOUND 2,I*10,10,14-I/2
980 FOR II=L TO 5:NEXT II:POKE 710,I*1
6:NEXT I:SC=SC+120:GOSUB 790
990 COLOR 32:PLOT XI,YI:XI=L:YI=L:POKE
 710,218:SOUND 2,L,L,L:KR=UK:GOTO 90
1000 REM [illegible]
1010 RESTORE 10300:A1=120:A2=136
1020 FOR I=33690 TO 33782 STEP 2:POKE
I,1:POKE I+1,2:NEXT I
1030 FOR I=33620 TO 33000 STEP 2:POKE
I,L:POKE I+1,2:NEXT I
1040 FOR I=39380 TO 39457:READ C:POKE
I,C:NEXT I:RESTORE 10700
1050 FOR I=39540 TO 39697:READ C:POKE
I,C:NEXT I
1070 CH=144:CC=CH*256:RESTORE 10800
1080 FOR I=1536 TO 1546:READ C:POKE I,
C:NEXT I:U=USR(1540):RESTORE 10400
1090 READ A:IF A=-1 THEN RETURN
1100 FOR I=L TO 7:READ B:POKE CC+A*8+I
,B:NEXT I:GOTO 1090
1150 REM [illegible]
1200 A1=120:A2=156
1210 FOR I=1049 TO 1999 STEP 16
1220 RESTORE 1380+(INT(RND(L)*5))*10
1230 FOR II=L TO 10:READ C:POKE 708,II
*10:POKE A2*256+II+1,C:POKE A2*256+II+
1,C
1240 NEXT II:NEXT I
1250 RESTORE 1380+2*10
1260 FOR I=L TO 9:READ C:POKE 38912+I,
C*16+42:IF C=L THEN POKE 38912+I,1
1270 POKE 38933+I,INT(RND(L)*16)*16+10
:POKE 38953+I,L:POKE 38923+I,1
1300 POKE 38943+I,C+1*20:NEXT I
1310 RETURN
```

```
1350 DATA 0,0,0,1,0,0,0,0,5,0
1360 DATA 0,0,0,0,0,0,0,0,0,17
1370 DATA 0,0,0,0,9,0,0,9,11,0
1380 DATA 0,0,13,4,0,15,0,0,0,0
1390 DATA 5,0,12,0,0,9,0,0,11,6
1400 DATA 0,7,0,17,10,16,0,5,0,11
1410 DATA 0,9,9,15,10,0,12,0,11,17
1420 DATA 0,1,12,7,0,0,7,7,12,0
1430 DATA 10,10,0,10,4,3,4,10,9,11
1440 DATA 9,15,0,14,1,15,9,17,2,13
1450 DATA 10,2,9,10,15,14,0,7,6,14
1460 DATA 14,0,14,0,0,6,10,2,3,17
1470 DATA 2,6,15,5,5,17,2,5,11,15
1480 DATA 13,5,12,5,2,15,10,9,17,15
1490 DATA 0,12,11,0,10,11,13,17,12,0
1500 DATA 11,7,9,17,10,16,15,0,17,11
1510 DATA 17,5,5,10,10,17,17,7,11,17
1520 DATA 11,4,12,7,17,8,14,7,12,4
1530 DATA 10,16,16,16,1,1,4,10,9,11
2000 REM [illegible]
2010 SOUND 1,L,L,L:POKE 534,1:SOUND L,
L,L,L:POKE 53251,L
2020 FOR I=100 TO 170 STEP 2:SOUND 2,I
,10,10:POKE 54279,132
2030 SOUND 2,201-I,10,10:POKE 54279,13
0:NEXT I:SOUND 2,L,L,L
2040 FOR I=L TO 4:POKE 53248+I,L:NEXT
I:FOR I=L TO 200:NEXT I
2050 SOUND 1,L,L,L:LB=LB-1:IF LB=L THE
N 3000
2060 GRAPHICS 18:GOSUB 1250:ZL=4:XP=120
:KR=L:YP=120:GOSUB 7020:GOTO 30
4000 REM [illegible]
4005 POKE 540,1:FOR I=L TO 4:POKE 5324
8+I,L:NEXT I:POKE 54286,250
4010 POKE 82,L:GRAPHICS L:POKE 559,L:P
OKE 752,1:POKE 709,202:POKE 708,184:PO
KE 711,236:POKE 710,220:POKE 756,CD
4030 DL=PEEK(560)+256*PEEK(561):POKE D
L+6,7:POKE DL+9,6:POKE DL+20,6
4040 ? "..........................
...........";
4050 ? CHR$(144);"  w o n d e l  ",
CHR$(144);
4060 ? "..........................
...........";
4070 ? CHR$(144),"(C)BY J/HC L/EFLER",
CHR$(144);
4080 ? "..........................
...........";
4090 ? "
      ";
4100 FOR I=L TO 7:? " ";I+1;". ",
4110 ? N$(I*20+1,I*20+12);" .Level.
. .";:POSITION 25,I*2+5:? NS(I+1
);
4120 IF PA(I+1)<>HE(I+1) THEN ? "-";SE
(I+1);
4130 POSITION (30-LEN(STR$(P(I+1)))),I
*2+5:? P(I+1);" ";
4140 ? "
      ";:NEXT I
4150 ? "......[illegible]
......";
4160 POKE 53277,1:POKE 704,12:POKE 745
,40:POKE 54279,136
4200 FOR I=L TO 3:SOUND I,L,L,L:NEXT I
:POKE 559,62
4210 RESTORE 4400+INT(RND(L)*8)*10
4220 READ A,B,C:SOUND 1,A,10,10:SOUND
2,B,10,8:SOUND 3,C,10,8
4230 FOR J=1 TO 70:IF STRIG(L) THEN NE
XT J:GOTO 4210
4240 FOR I=170 TO 190 STEP -0.2:POKE 5
3761,I:POKE 53760,I:POKE 53767,I:NEXT
I
4270 IF OE=1 THEN GOSUB 1250:RETURN
4280 X=19:Y=15:GOSUB 4390:X=32:Y=17:GO
SUB 4390:X=32:Y=19:GOSUB 4390:X=19:Y=2
1:GOSUB 4390
4290 RN=1:POSITION 2,15:? "X Press STA
RT to begin a new game! X",:POKE 752,L
4300 POSITION 2,17:? "Press SELECT to
choose 1st LEVEL ";CHR$(160),CHR$(172)
;
4310 IF PEEK(53279)=5 THEN GOSUB 4480
```

```
4320 IF PEEK(53279)=6 THEN POKE 559,L:
GOTO 4350
4340 GOTO 4310
4350 R1=R1:R2=R2:RA=RB:VL=0.78-RR*0.01
:VE=60-RR*2:VZ=6-RR*0.25:GOSUB 1200
4370 RETURN
4380 DD=DD+1:IF DD=10 THEN DA=1
4385 POSITION 38,17:? CHR$(RR+178);:FO
R I=250 TO 60 STEP -15:SOUND 2,I,10,10
:NEXT I:SOUND 2,L,L,L:RETURN
4390 POSITION X,Y:FOR I=L TO 37:? CHR$
(X);:SOUND 2,I,10,10:SOUND 2,L,L,L:NEX
T I:RETURN
4400 DATA 96,121,102
4410 DATA 128,91,81
4420 DATA 121,96,81
4430 DATA 144,121,96
4440 DATA 108,91,72
4450 DATA 102,144,121
4460 DATA 121,102,193
4470 DATA 128,91,81
4480 DATA 144,193,243
4490 DATA 121,144,102
4500 DATA 128,102,182
4700 REM [illegible]
4701 TI=(80-PEEK(19))*100:POKE 53252,L
4705 POKE 530,L:SOUND L,L,L,L:FOR I=10
 TO L STEP -1:FOR II=L TO 4:SETCOLOR I
I,I,I:SOUND 2,I*5+11,10,I:NEXT II
4710 NEXT I:FOR I=L TO 3:POKE 53248+I,
L:NEXT I:SOUND 2,L,L,L:GRAPHICS 17:POK
E 708,102
4720 POKE 709,122:POKE 710,L:POKE 559,
34:DL=PEEK(560)+256*PEEK(561)
4730 POKE DL+8,7:POKE DL+9,7:POKE DL+1
1,2:POKE DL+22,2
4740 POSITION 4,3:? #6;"w o m b e l"
4750 POKE 708,206:POSITION 7,6:? #6;"(
C) 1986 by J";CHR$(12);"OE L";CHR$(12)
;"FFLER":TT=INT(TI)*100
4760 POSITION 4,11:? #6;"LEVEL: ";RR
4770 POSITION 4,13:? #6;"SCORE: ";SC
4780 POSITION 4,15:? #6;"bonus: ";TT
4800 SOUND L,L,L,L:POSITION 6,19:? #6;
"PUSH BUTTON TO CONTINUE"
4810 VE=L:BL=4:X5=10:RO=L:SC=SC+TT:RR=
RR+1:R1=R1+1:R2=R2+1:IF R1>11 THEN R1=
1
4820 IF R2=20 THEN R2=10
4830 VL=VL-0.01:IF VL<0.3 THEN VL=0.3
4840 VE=VE-3:IF VE<20 THEN VE=20
4850 VZ=VZ-0.25:IF VZ<2 THEN VZ=2
4870 RET=1:GOSUB 4200:GOSUB 7020:GOTO
20
5000 REM [illegible]
5010 GRAPHICS 18:POKE 559,L:POKE 710,1
24:POKE 709,20:POKE 711,120:POKE 752,1
:POKE 623,2
5020 DL=PEEK(560)+256*PEEK(561)-1:POKE
 560,PEEK(560)-1:POKE DL+8,7
5030 POKE DL,112:POKE DL+3,66:POKE DL+
4,PEEK(88):POKE DL+5,PEEK(89):POKE 756
,CH:POKE 87,1
5035 FOR I=DL+6 TO DL+15:POKE I,PEEK(I
)+128:NEXT I:POKE 512,0:POKE 513,152:
POKE 559,42:POKE 1761,164
5040 POSITION 1,L:? #6;"SCORE ";SC:POS
ITION 14,L:? #6;"LEVEL ";RR
5050 POSITION 3,1:? #6;"TIME 50":POSIT
ION 12,1:? #6;"LIVES:";LB
5060 COLOR 219:PLOT L,2:COLOR 220:PLOT
 1,2:DRAWTO 18,2:COLOR 221:PLOT 19,2:R
ESTORE 5000+RR*100
5070 COLOR 172:PLOT 19,2:DRAWTO 19,12:
COLOR 222:PLOT 19,13:COLOR 220:PLOT 18
,13:DRAWTO 1,13:COLOR 223:PLOT L,13
5071 COLOR 172:PLOT L,12:DRAWTO L,3:CO
LOR 171:PLOT X5,2:PLOT X5,13:COLOR 139
5075 READ X,Y:IF X=99 THEN POKE 54286,
192:GOSUB 6500:RETURN
5080 IF X=L THEN COLOR Y:GOTO 5075
5085 IF X<L THEN PLOT ABS(X),Y:GOTO 50
75
5090 DRAWTO X,Y:GOTO 5075
5095 REM LEVEL 1
5100 DATA -10,12,16,12,16,6,12,6,-12
5110 DATA 12,14,16,3,12,0,12,2,5,2
5120 DATA 10,0,10,5,0,10,0,10,11,-5
5130 DATA 3,5,7,7,7,7,4,10,0,10,5,10
5140 DATA 6,10,0,0,2,1,3,1,0,-6,7,7
5150 DATA 7,-11,5,13,5,-1,11,-11,7,12
5160 DATA 8,17,5,17,11,11,12,9,12,0
5170 DATA 17,7,11,11,0,16,4,99
5195 REM LEVEL 2
5200 DATA -1,3,4,3,4,4,6,4,6,6,1,6
5210 DATA 1,8,3,8,-2,12,6,12,5,10,9
5220 DATA 16,9,11,-10,12,14,10,17,10
5230 DATA -6,0,18,0,18,4,19,4,17,3,10
5240 DATA 3,10,7,8,2,7,9,10,9,13
5250 DATA 12,11,12,-6,12,8,11,-8,7,4
5260 DATA 7,-1,5,1,4,-7,1,9,1,16,3
5270 DATA 16,5,12,5,17,7,8,37,15,8
5280 DATA 10,11,4,18,99
5295 REM LEVEL 3
5300 DATA -1,12,18,3,18,11,18,12,-4
5310 DATA 12,7,12,15,12,9,12,9,9,12
5320 DATA 9,12,8,8,0,-8,3,9,2,-2,4,2
5330 DATA 8,4,8,-15,4,17,4,17,7,6,2
5340 DATA 4,3,5,3,-3,8,3,7,5,7,-5,11
5350 DATA 8,11,8,8,9,11,5,11,-12,5,14
5360 DATA 6,14,11,10,11,8,37,-17,4,-7
5370 DATA 9,-7,8,99
5395 REM LEVEL 4
5400 DATA 2,8,2,3,10,3,-4,5,5,5,-2,9
5410 DATA 4,9,4,11,2,11,2,10,-8,10,18
5420 DATA 10,-8,6,10,6,10,8,8,8,8,7
5430 DATA -9,12,-11,12,-13,12,-15,10
5440 DATA 18,10,18,8,-15,4,17,4,17,6
5450 DATA 15,6,16,6,8,2,1,8,7,8,7,5
5460 DATA 10,3,12,7,16,7,17,12,10
5470 DATA 17,1,12,7,12,8,37,16,5,-9
5480 DATA 7,3,10,99
5495 REM LEVEL 5
5500 DATA -6,3,8,4,10,4,10,3,-18,3,14
5510 DATA 5,10,5,10,3,1,4,5,4,5,7,1
5520 DATA 7,1,5,-2,8,2,11,1,12,3,12,3
5530 DATA 12,7,8,-12,7,17,7,17,9,15,9
5540 DATA 15,11,10,11,0,9,8,12,10,9
5550 DATA 10,17,8,2,-1,3,3,3,-9,3,-11
5560 DATA 3,13,3,-15,3,15,4,-2,6,-4,6
5565 DATA -8,8,8,6,4,8,3,9,3,10,9,9
5570 DATA -7,11,-9,12,-11,12,10,12,-8
5575 DATA 8,-11,6,15,6,-17,5,17,6,13
5580 DATA 8,15,8,8,37,16,8,-6,6,-9
5590 DATA 11,99
5595 REM LEVEL 6
5600 DATA 1,4,5,4,5,9,7,9,7,6,12,6
5610 DATA -10,3,15,3,15,8,10,10,15
5620 DATA 10,11,10,13,11,11,11,11,8
5630 DATA 5,8,9,11,3,11,3,8,1,8,-3,12
5640 DATA 8,12,8,2,-10,12,-9,12,-9
5650 DATA 10,-9,12,-2,12,1,8,1,10
5660 DATA 3,8,3,10,5,3,-11,4,11,5
5670 DATA 10,8,-13,3,13,5,18,5,8,17
5680 DATA 12,11,10,4,-2,9,99
5695 REM LEVEL 7
5700 DATA -1,12,5,12,5,10,3,10,3,8,8
5710 DATA 8,8,6,2,6,1,3,14,3,10,7,10
5720 DATA 11,10,12,-9,12,9,8,12,8,12
5730 DATA 4,-17,12,18,12,14,10,-16,4
5740 DATA 18,4,18,8,8,2,-15,4,15,8,10
5750 DATA 8,10,4,7,4,7,5,4,5,-18,7
5760 DATA 10,7,-5,7,7,7,2,13,2,9,-7
5770 DATA 9,8,9,-8,11,7,11,7,12,-10
5780 DATA 18,10,13,8,37,3,4,-6,9,-10
5790 DATA 10,99
5795 REM LEVEL 8
5800 DATA -8,4,16,8,10,12,7,12,3,8,7
5810 DATA 8,11,8,8,11,-2,12,2,4,-17,8
5820 DATA 10,4,17,12,16,12,1,8,8,2
5830 DATA -5,3,9,3,-16,3,18,3,-18,7
5840 DATA 17,8,10,9,-17,8,15,8,-3,5,8
5850 DATA 5,4,6,10,18,10,12,14,5,10
5860 DATA 8,10,1,-1,9,1,12,-5,8,5,11
5870 DATA 8,37,3,4,-17,6,-17,10,99
5895 REM LEVEL 9
5900 DATA -4,6,4,3,1,3,1,12,8,12,4,10
5910 DATA -6,4,6,12,-9,8,9,9,-11,4,11
5920 DATA 11,12,12,14,12,14,8,10,3
5930 DATA 10,3,10,12,10,12,10,7,8,2
5940 DATA -7,10,-10,11,-12,12,-5,8,-7
5950 DATA 7,7,3,5,3,-13,4,13,7,12,7
5960 DATA 12,9,13,9,13,10,-17,7,-15
5970 DATA 10,15,12,-2,4,2,4,8,8,8,37
5980 DATA 2,11,10,7,-17,11,99
5995 REM LEVEL 10
```

```
6000 DATA 3,6,2,0,0,4,0,7,0,11,9,11
6010 DATA 3,7,-12,4,12,11,16,11,-16,3
6020 DATA 18,3,18,0,18,4,0,2,-16,7,17
6030 DATA 0,-12,6,12,7,-12,18
6040 DATA -7,4,3,12,4,6,7,0,3,12,6
6050 DATA 12,-1,3,1,0,-15,12,16,12,0
6060 DATA 17,-7,7,7,11,18,11,99,0
6500 COLOR 160:PLOT 12,10
6510 XT=INT(RND(L)*18)+1:YT=INT(RND(L)*18)+3
6520 LOCATE XT,YT,LD:IF LD<>32 THEN 6510
6530 COLOR 44:PLOT XT,YT:COLOR 32:PLOT 12,10:RETURN
7000 REM ANFANGSBEDINGUNGEN
7010 GET L:CH=144:GOSUB 8000:SC=L:BB=L:ZL=4:LD=1:XS=10:POKE 1781,164
7020 GOSUB 5000:POKE 652,L:POKE 623,0:POKE L,L:POKE 1,16:POKE 54279,118:XP=144:YP=107:POKE 766,L
7030 SOUND L,L,10,8:POKE 203,XP:POKE 208,YP:POKE 206,L:POKE 207,L:POKE 53256,128:POKE 53257,L:POKE 53277,3
7040 POKE 53278,L:POKE 558,1:POKE 17,L:POKE 16,L:POKE 53770,1
7050 RETURN
9000 REM GAME OVER
9005 POKE 54286,255:TRAP 40000
9010 ? CHR$(125):GRAPHICS L:POKE 765,L:POKE 710,L:POKE 708,122:POKE 711,202
9015 DL=PEEK(560)+256*PEEK(561)
9020 POKE DL+6,6:POKE DL+7,6:POSITION 0,3:? #6,"wwwwW GAME   wwwwW":POSITION 29,1:? #6,"OVER".?
9030 POSITION 13,7:? #6,"YOUR SCORE:",SC:IF P(7)>SC THEN 9300
9040 POSITION 9,10:? "PLEASE ENTER YOUR NAME ":POKE 730,66
9050 POKE DL+17,6:POKE DL+20,6:POKE DL+21,7:POKE DL+22,6
9055 COLOR 251:PLOT 23,13:COLOR 252:PLOT 24,13:DRAWTO 35,13:COLOR 253:PLOT 36,13
9060 POSITION 3,14:? #6,CHR$(160):POSITION 16,14:? #6,CHR$(160)
9065 COLOR 255:PLOT 23,14:COLOR 252:PLOT 24,14:DRAWTO 35,14:COLOR 254:PLOT 36,14
9070 POSITION 5,16:? "Press RETURN for a new game!"
9080 X=4:OPEN #2,4,L,"K:":SP$="........
"
9090 SOUND 2,L,L,L:GET #2,G
9100 IF G=155 THEN FOR I=210 TO L STEP -10:SOUND 2,I,10,10:NEXT I:GOTO 9140
9110 IF G=126 AND X>4 THEN SOUND 2,60,10,6:X=X-1:POSITION X,14:? #6,CHR$(32);:SP$(X-3,X-3)=".":GOTO 9090
9120 IF G<32 OR G>90 OR X>18 THEN SOUND 2,120,L,10:GOTO 9090
9130 POSITION X,14:? #6;CHR$(G);:SOUND 2,121,10,10:X=X+1
9137 SP$(X-4,X-4)=CHR$(G):GOTO 9090
9140 CLOSE #2:FOR I=1 TO 8:IF SC>P(I) THEN 9160
9150 NEXT I:GOTO 9300
9160 X=77:Y=100:FOR II=8 TO I STEP -1:P(II+1)=P(II):NA(II+1)=NA(II):BE(II+1)=BE(II):HI$(X+12,Y+12)=HI$(X,Y)
9170 X=X-12:Y=Y-12:NEXT II:P(I)=SC:NA(I)=AA:BE(I)=BB
9180 HI$(X+12,Y+12)=HI$(X,Y):HI$(X,Y)=SP$
9290 GOTO 7005
9300 POKE DL+18,7:POKE DL+19,6:POSITION 0,12:? #6,"PUSH BUTTON!"
9310 POSITION 9,10:? "High Scorer Nr.1:",P(1)
9320 IF STRIG(L) THEN 9320
9400 POP :POP :GOTO 11
10000 REM [illegible]
10010 DATA 91,76,91,121,144,121,102,0,91,76,91,123,144,121,102,0
10020 DATA 91,81,76,81,76,91,81,91,81,102,91,121,144,121,102,0
10030 DATA 91,81,76,81,76,91,81,91,81,102,91,121,144,121,102,0
10040 POKE 709,100:RESTORE 10010:FOR I=37380 TO 37767:READ C:POKE I,C:NEXT I
10070 REM [illegible]
10100 RESTORE 10110:FOR I=1536 TO 1791:READ C:POKE I,C:NEXT I:RETURN
10110 DATA 169,2,141,26,2,173,4,210,201,0,208,100,173,5,210,201,0,208,183,160,0,165,1,133,130
10120 DATA 0,220,0,208,7,162,0,134,0,173,120,2,133,1,201,7,208,11,238,201,182,136,162,7,212
10130 DATA 162,1,134,209,201,11,208,11,198,201,162,120,162,7,212,162,2,134,203,201,14,208,8,170,206
10140 DATA 178,206,162,2,134,209,201,13,208,8,238,208,210,204,162,4,134,209,169,0,162,0,164,206,153
10150 DATA 0,132,153,0,140,151,0,133,153,0,141,232,200,224,17,208,238,76,123,6,76,163,6,162,0
10160 DATA 164,206,189,172,153,153,0,132,169,212,153,153,0,140,189,232,153,153,0,132,189,252,153,153,0
10170 DATA 141,232,200,224,17,208,238,165,204,133,205,165,201,141,0,208,141,1,208,167,0,162,0,164,208
10180 DATA 183,0,124,183,0,142,232,200,224,17,208,244,162,0,160,207,189,16,154,153,0,134,153,0,142
10190 DATA 232,200,224,17,208,201,165,207,133,200,165,208,141,2,208,238,196,2,206,38,153,171,30,153,201
10200 DATA 164,208,24,169,167,141,10,182,174,31,167,189,132,153,141,0,210,232,220,40,208,2,162,0
10210 DATA 142,31,162,32,32,151,76
10290 REM [illegible]
10300 DATA 12,18,27,30,20,12,12,20,24,24,60,50,55,31,3,3,0,0,0,0
10310 DATA 48,120,104,120,120,48,60,56,24,24,76,100,238,240,120,120,0,0,0,0
10320 DATA 0,0,32,30,0,120,120,174,102,30,12,0,0,0,36,190,0,0,0,0
10330 DATA 0,0,4,6,0,1,1,87,101,100,48,16,16,0,26,54,0,0,0,0
10340 DATA 62,62,127,127,127,127,127,127,127,127,127,127,127,62,62,0,0,0,0
10350 DATA 0,0,0,24,30,92,63,133,219,255,255,255,226,126,60,0
10360 DATA 0,24,60,126,90,90,219,219,255,183,153,153,107,60,231,0
10370 DATA 0,0,24,60,120,255,153,219,255,255,219,90,102,60,231,0
10380 DATA 0,24,60,126,219,153,255,153,102,60,24,60,102,195,129,0
10390 DATA 0,231,189,165,231,231,24,24,60,102,219,187,219,102,126,0
10400 DATA 7,0,62,127,127,127,127,127,63
10410 DATA 12,20,30,90,40,20,40,30,40
10420 DATA 15,0,102,0,60,102,102,60,0
10430 DATA 30,0,0,65,170,65,170,0,0
10440 DATA 63,20,40,21,42,21,42,0,0
10450 DATA 57,0,0,21,42,21,42,30,40
10460 DATA 62,0,0,84,168,84,168,20,40
10470 DATA 32,0,0,0,20,62,20,6,0
10480 DATA 62,20,40,84,168,84,168,0,0
10490 DATA 27,36,25,64,97,5,32,26,26
10500 DATA 30,24,27,80,6,80,70,20,60
10510 DATA 29,0,34,21,76,20,33,10,13
10520 DATA 10,0,64,24,0,64,10,0,16
10530 DATA 11,0,126,30,126,60,102,126,0
10540 DATA 5,0,62,127,107,93,107,127,62
10550 DATA 2,0,56,124,98,123,30,60,0
10600 DATA -1
10700 DATA 24,60,84,60,60,12,12,24,24,10,58,55,62,125,1,0,0,0,0,0
10710 DATA 24,60,52,60,60,40,40,24,24,72,76,236,252,176,120,0,0,0,0,0
10720 DATA 0,0,64,192,0,0,2,6,70,100,200,136,120,0,10,204,0,0,0,0
10730 DATA 0,0,2,3,0,0,64,96,100,50,51,17,1,0,16,51,0,0,0,0
10740 DATA 67,82,119,119,107,107,93,93,107,107,119,119,62,82,0,0,0,0
10800 DATA 104,162,0,189,0,224,157,0,144,189,0,225,157,0,165,189,0,226
```

```
10810 DATA 157,0,166,109,0,227,157,0,1
17,227,224,255,200,227,96
10900 REM DLIST-INTERRUPT
10910 POKE 788,156:RESTORE 11000:FOR I
=15400 TO 29142:READ C:POKE I,C:NEXT I
10920 RESTORE 11100:FOR I=39200 TO 392
68:READ C:POKE I,C:NEXT I:RETURN
11000 DATA 72,138,72,152,72,120,174,10
,152,189,0,152,201,0,240,97,189,21,152
,111,21
11010 DATA 200,189,11,152,201,1,208,6,
254,0,152,76,126,152,222,0,152,204,41,
152,189
11020 DATA 41,152,201,0,208,67,169,0,1
57,41,152,189,11,152,201,1,240,16,222,
31,152
11030 DATA 222,11,152,189,31,152,204,3
1,152,76,175,152,204,31,152,254,31,152
,189,31,152
11040 DATA 222,11,152,160,189,172,160,
201,0,240,20,189,11,152,201,3,208,0,16
9,1,157
11050 DATA 11,152,70,203,152,169,0,157
,11,152,189,0,152,141,3,208,238,10,152
,173,10
11060 DATA 152,201,10,208,5,169,0,141,
10,152,104,168,104,170,104,64
11100 DATA 172,30,192,141,1,210,169,0,
141,10,152
11110 DATA 169,0,141,208,2,173,120,2,2
01,15,208,1,96,206,22,193,173,22,153,2
01,0,208,15,169,4
11120 DATA 141,22,153,162,0,189,192,15
1,72,189,76,154,157,192,153,104,157,76
,154,232,224,79,208,237,96
12000 REM INITIALISIERUNG
12010 DIM SP$(20),MI$(120),P(10),RD(10
),RE(10):FOR I=1 TO 3:POKE 53760+I,1:N
EXT I
12020 GRAPHICS 21:POKE 559,1:POKE 88,1
:POKE 89,126:? 00;CHR$(125):POKE 53760
,24:POKE 53761,164:POKE 53762,207
12030 POKE 53765,168:POKE 53767,162:PO
KE 53768,200:POKE 53762,58:POKE 53764,
270:POKE 53766,101
12040 GRAPHICS 1:POKE 710,1:POKE 756,2
04:POKE 755,1:POKE 708,122
12050 DL=PEEK(560)+256*PEEK(561):POKE
DL+3,6:POKE DL+14,7:POKE DL+20,6
12060 POSITION 2,4:? "please wait for"
:POSITION 24,6:? "M O R D E L"
12070 POSITION 2,14:? "(C) 1986 by J";
CHR$(123);"RG L";CHR$(123);"FFLER"
12100 FOR I=1 TO 8:MI$(I*12+1,I*12+12)
="white MORDEL":NEXT I
12110 FOR I=1 TO 8:P(I)=5500-I*100:RD(I
)=1:RE(I)=2:NEXT I
12120 GOSUB 1000:GOSUB 10000:GOSUB 109
10:GOSUB 1700:RETURN
12400 REM [illegible]
```

Diskeditor

Mit diesem Diskeditor für alle Atari-Computer kann beliebig auf Sektoren der Diskette im Laufwerk 1 zugegriffen werden. Nach dem Start erscheint folgendes Menü:

xxxx R Sektor xxxx lesen
C angezeigten Sektor verändern
W angezeigten Sektor schreiben
+ Sektor +1 lesen
– Sektor –1 lesen
A ASCII-Darstellung
H HEX-Darstellung
? Kommandos anzeigen

Zu den einzelnen Menü-Punkten:

xxxx R: Mit dieser Option wird ein beliebiger Sektor gelesen und angezeigt (z. B. liest 361 R den ersten Sektor des Directory einer DOS-II Diskette).

C: Hiermit kann man mit dem Cursor auf dem Bildschirm umherfahren und den angezeigten Sektor verändern. Je nach Modus werden nur die Zeichen 0-9 und A-F oder alle ASCII-Zeichen entgegengenommen. Mit RETURN gelangt man in den normalen Kommando-Modus zurück. Alle auf dem Bildschirm vollzogenen Änderungen wurden eingelesen, aber noch nicht auf die Diskette zurückgeschrieben.

W: Das Zurückschreiben eines veränderten Sektors geschieht mit diesem Befehl. Vorsicht: der Befehl wird ohne Rückfrage ausgeführt.

+ –: Mit diesen Kommandos ist es möglich, vorwärts und rückwärts zu "blättern". Als Kurzform für + RETURN ist auch RETURN ohne Eingabe zulässig.

A: Umschaltung auf ASCII-Darstellung. Alle Bytes eines Sektors werden als ASCII-Zeichen dargestellt. Statt RETURN (155) wird ein SPACE ausgegeben. Soll ein RETURN-Zeichen eingefügt werden, so ist dies nur im Hex-Mode möglich. Nach dem Starten befindet man sich automatisch im ASCII-Mode.

H: Umschalten auf hexadezimale Darstellung der Bytes.

?: Anzeige des Menüs, es kann auch HELP eingegeben werden.

Anwendungsbeispiel: Mit diesem Programm ist es möglich, gelöschte Files wieder lesbar zu machen. Das Directory einer DOS II Diskette befindet sich ab Sektor 361. Vor dem Filenamen werden 5 Byte mit der folgenden Bedeutung abgelegt. Das erste Byte ist das Statusbyte. Es gibt an, ob das File gelockt, gelöscht usw. ist. Byte 2 und 3 geben die erste Sektornummer des Files an, Byte 4 und 5 seine Länge. Bei einem gelöschten File hat das erste Byte den Wert 128. Setzt man ihn auf 98, so ist das File wieder lesbar, sofern es nicht überschrieben wurde.

Herbert Bartk

Disk-Editor

```
100 REM
110 REM DISKEDITOR
120 REM
130 DIM ML$(37),A$(128),CMD$(7)
140 DIM HEX$(16)
150 HEX$="0123456789ABCDEF"
160 A$(128)="E":SUMM=1:ASCII=SUMM
170 OPEN #1,4,0,"E:":OPEN #2,4,0,"K:"
180 FOR I=1 TO 37:READ A:ML$(I)=CHR$(A
):NEXT I
190 FOR I=1770 TO 1815:READ A:POKE I,A
:NEXT I
200 REM
210 REM [illegible]
220 REM
230 GRAPHICS 0:POKE 82,1:POKE 83,39
240 POKE PEEK(560)+256*PEEK(561)+25,11
2
250 POKE 512,234:POKE 513,6
260 POKE 54286,192:PRINT :PRINT
270 PRINT "XXXX R: Sektor XXXX lesen"
280 PRINT "      C: angezeigten Sektor
veraendern"
290 PRINT "      W: angezeigten Sektor
schreiben"
300 PRINT "      +: Sektor+1 lesen"
310 PRINT "      -: Sektor-1 lesen"
320 PRINT "      A: ASCII Darstellung"
330 PRINT "      H: HEX Darstellung"
340 PRINT "      ?: Kommandos anzeigen"
350 REM
360 REM KOMMANDO ABFRAGEN
370 REM
380 POSITION 5,22:PRINT "?";
390 PRINT "KOMMANDO: ";
400 POKE 694,0:POKE 702,64:TRAP 380
410 CMD$="":INPUT #1,CMD$
420 IF LEN(CMD$)=0 THEN 380
430 IF CMD$="?" OR CMD$="HELP" THEN 84
0
```

```
440 IF CMD$="A" AND ASCII=0 THEN 500
450 IF CMD$="H" AND ASCII=1 THEN 600
460 IF CMD$="+" OR CMD$="" THEN 820
470 IF CMD$="-" THEN 850
480 IF CMD$="W" THEN 310
490 IF CMD$(LEN(CMD$))="R" THEN IF LEN
(STR$(VAL(CMD$)))=LEN(CMD$)-2 AND CMD$
(LEN(CMD$)-1,LEN(CMD$)-1)=" " THEN 750
500 IF CMD$="C" AND ASCII=1 THEN 960
510 IF CMD$="C" AND ASCII=0 THEN 1110
520 GOTO 400
530 REM
540 REM ASCII-DARSTELLUNG
550 REM
560 ASCII=1:POKE 82,4:POKE 83,39
570 PRINT "K":POSITION 4,1
580 PRINT "SEKTORNUMMER: ";SNUM
590 PRINT
600 FOR I=1 TO 128
610 A=ASC(A$(I)):IF A=155 THEN A=32
620 PRINT "";CHR$(A);:NEXT I:GOTO 300
630 REM
640 REM HEX-DARSTELLUNG
650 REM
660 ASCII=0:IF A=0 THEN 350
670 PRINT "K":POSITION 8,1
680 PRINT "SEKTORNUMMER: ";SNUM
690 POKE 82,8:POKE 83,31:PRINT
700 FOR J=1 TO 128:I=ASC(A$(J,J)):A=IN
T(I/16)+1:? HEX$(A,A);:A=(I/16-INT(I/1
6))*16+1:? HEX$(A,A);" ";:NEXT J
710 POKE 82,2:GOTO 300
720 REM
730 REM SEKTOR LESEN
740 REM
750 SNUM=VAL(CMD$)
760 CMD=82:GOSUB 1350
770 IF ASCII=1 THEN 560
780 GOTO 660
790 REM
800 REM SEKTOR+1 LESEN
810 REM
820 SNUM=SNUM+1:GOTO 760
830 REM
840 REM SEKTOR-1 LESEN
850 REM
860 SNUM=SNUM-1:IF SNUM<1 THEN SNUM=1
870 GOTO 760
880 REM
890 REM SEKTOR SCHREIBEN
900 REM
910 IF R<>0 THEN CMD=87:GOSUB 1350
920 GOTO 300
930 REM
940 REM ASCII-AENDERN
950 REM
960 POSITION 1,22:PRINT "AENDERN hit
RETURN to quit !";
970 POSITION 6,2:PRINT "++";
980 GET #2,I:PRINT CHR$(I);
990 IF I=155 THEN 1030
1000 IF PEEK(84)<2 THEN ? "↓↓↓↓";
1010 IF PEEK(84)>8 THEN ? "↑↑↑↑";
1020 GOTO 980
1030 POKE PEEK(94)+256*PEEK(95),PEEK(9
3)
1040 POSITION 6,1:FOR I=1 TO 128
1050 GET #6,A:IF ASC(A$(I))<>155 OR A<
>32 THEN A$(I,I)=CHR$(A)
1060 NEXT I
1070 PRINT "++";A$(128):GOTO 300
1080 REM
1090 REM HEX-AENDERN
1100 REM
1110 POSITION 1,22 PRINT "AENDERN hit
RETURN to quit !";
1120 POKE 82,0:POKE 83,36
1130 POSITION 8,3:PRINT "++";
1140 GET #2,I:IF I=155 THEN 1230
1150 IF I<28 OR I>70 THEN 1140
1160 IF I<48 AND I>31 THEN 1140
1170 IF I<65 AND I>57 THEN 1140
1180 PRINT CHR$(I);:IF PEEK(93)=26 AND
 I<>30 THEN PRINT "+";
1190 IF PEEK(93)=26 AND I=30 THEN PRIN
T "+";
1200 IF PEEK(93)<>0 THEN 1140
1210 I=PEEK(84)
1220 IF I=17 AND PEEK(85)>32 THEN POSI
TION 1,12:PRINT "++";
1230 IF I=2 THEN ? "↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓";
1240 IF I=19 THEN ? "↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑"
;
1250 GOTO 1140
1260 POKE PEEK(94)+256*PEEK(95),PEEK(9
3)
1270 POSITION 8,1:FOR J=1 TO 128
1280 GET #6,A:A=A-48:IF A>9 THEN A=A-7
1290 GET #6,I:I=I-48:IF I>9 THEN I=I-7
1300 A$(J,J)=CHR$(I+A*16):GET #6,A:NEX
T J
1310 PRINT "++":POKE 82,2:POKE 83,39:
GOTO 300
1320 REM
1330 REM DISK HANDLER
1340 REM
1350 R=USR(ADR(HL$),CMD,ADR(A$),SNUM)
1360 POKE 54286,192
1370 IF R=1 THEN RETURN
1380 IF R=139 AND SNUM=721 OR SNUM<1 THEN SNUM=SNUM-1
1390 POP :POSITION 2,22:PRINT "";
1400 PRINT "ERROR ";R
1410 FOR I=1 TO 350:NEXT I:GOTO 230
1420 REM
1430 REM DATA FUER DISK HANDLER
1440 REM
1450 DATA 104,169,1,141,1,3,104,104
1460 DATA 141,2,3,104,141,5,3,104
1470 DATA 141,4,3,104,141,11,3,104
1480 DATA 141,10,3,32,83,228,132,212
1490 DATA 169,0,133,213,96
1500 REM
1510 REM [illegible]
1520 REM
1530 DATA 72,169,230,141,10,212,141,24
1540 DATA 208,169,0,141,23,208,104,64
```

Calc 800

"Calc 800" ist ein universelles, allgemein verwendbares mathematisches Programm. Es erlaubt sowohl einfache wie auch relativ komplexe Probleme mathematischer, kaufmännischer und technischer Art mit dem Atari zu lösen. Es läßt sich auf Diskette abspeichern und bei Bedarf wieder laden. Mit der eingebauten Hardcopy-Funktion können beliebige Ausdrücke auf Atari- und Epson-kompatiblen Printern angefertigt werden.

Jedes Arbeitsblatt (Spreadsheet) besteht aus einer variablen Matrix mit maximal 2016 Einträgen. Über die jeweiligen Spalten- und Zeilennummern werden die Matrixfelder angesprochen und können mit mathematischen Funktionen miteinander verknüpft werden. Der Vorteil liegt nun darin, daß sämtliche Fixwerte und Berechnungsformeln nur einmal eingegeben werden müssen und damit beliebig viele Berechnungen möglich sind. Damit können verschiedene Modelle durchgespielt werden.

Das Programm beinhaltet zur leichten Handhabung eine HELP-Funktion, die jederzeit vom Eingabe-Modus aus anwählbar ist. Nach Benutzung von HELP wird wieder ins Kalkulationsprogramm zurückverzweigt und das Arbeitsblatt mit aktuellem Stand wieder aufgebaut.

Folgende Kommandos sind möglich:

Z	Fenster um eine Zeile abwärts
A	Fenster zum Blattanfang
E	Fenster zum Blattende
SL	Fenster um eine Spalte nach links schieben
SR	Fenster um eine Spalte nach rechts schieben
SH	Fenster um beliebige Spalten schieben
SV	Spaltenanzahl am Bildschirm ändern

- N – Spaltenname eingeben
- S – Spaltensumme errechnen
- SO – Sortierung einer Spalte
- W – feldweise Werteingabe
- WV – Wert eines Feldes ändern
- O – Ausstieg aus dem Programm
- LF – alle Matrixfelder löschen (Formeln bleiben erhalten)
- L – Felder und Formeln löschen (Neunumerierung)
- LA – Arbeitsblatt laden
- SP – Arbeitsblatt speichern
- HC – Hardcopy anfertigen
- RR – Rückrechnen, d. h. alle Feldinhalte sämtlicher Spalten werden nach Formel oder Wertänderung aktualisiert
- FE – Formeleingabe. Die berechneten Resultate werden in die jeweiligen Felder und Spalten eingetragen

Eine Formel besteht aus einem oder zwei Operanden, und bei jedem gibt es 3 Möglichkeiten:

a) eine Zahl: sie wird in jede Zeile der Spalte eingetragen

b) S und Spaltennummer. So trägt z. B. S3 die Summe der ersten Spalte in jede Zeile der angegebenen Spalte ein

c) ZS und Spaltennummer. ZS2 bildet z. B. die Zeilensummen ab Spalte 2

Mit zwei Operanden gilt das Format OPERAND1 OPERAND2 OPERATOR. Der erste Operand darf keine Zahl sein. Er muß K (Spaltennummer) oder S (Spaltennummer) lauten, also z. B. K3 und S2. Der zweite Operand kann eine Zahl oder K (Spaltennummer) oder S (Spaltennummer) sein. Als letzte folgt der Operator:

+ – *, / (Grundrechnungsarten), W (Wurzelziehen), ^(Potenzieren), % (prozentuale Differenz zwischen Op 1 und Op 2)

Zu trennen sind die Operanden und der Operator durch Komma, Strichpunkt oder Doppelpunkt (wahlfrei).

Hier einige Beispiele:

K1 K3 * multipliziert die Felder der Spalte 1 mit den Feldern der Spalte 3 und trägt die Ergebnisse in die Felder der angegebenen Spalte ein

K4 S2 + addiert zu jedem Feld der Spalte 4 die Spaltensumme der Spalte 2 und trägt die Ergebnisse in die Felder der angegebenen Spalte ein

K5 W berechnet die Wurzeln der Felder der Spalte 5 und trägt die Ergebnisse in die Felder der angegebenen Spalte ein

K10 2.5 / dividiert jedes Feld der Spalte 10 durch den Faktor 2.5 und trägt die Ergebnisse in die Felder der angegebenen Spalte ein

Achtung: Das Programm rundet alle Eingaben auf 2 Kommastellen ab.

Gerhard Leckel

```
KOMMANDO?  20 ZEILEN  5 SPALTEN  HILFE
=======S1===========S2===========S3=====
   BRUTTO          3% SKONTO    ZW1.SUMME
1  10              -0.3         9.7
2  20              -0.6         19.4
3  30              -0.9         29.1
4  40              -1.2         38.8
5  50              -1.5         48.5
6  60              -1.8         58.2
7  70              -2.1         67.9
8  80              -2.4         77.6
9  90              -2.7         87.3
10100              -3           97
11110              -3.3         106.7
12120              -3.6         116.4
13130              -3.9         126.1
14140              -4.2         135.8
15150              -4.5         145.5
16160              -4.8         155.2
17170              -5.1         164.9
  2100             -63          2037

EINGABE^HC
```

Bild 1 Hardcopy

Calc 800

```
10 REM [illegible] | [illegible]
[illegible]
100 SETCOLOR 2,9,0
105 GRAPHICS 2
106 POSITION 1,0:? #6;"      [illegible]"
107 POSITION 1,1:? #6;"      [illegible]"
108 POSITION 1,3:? #6;" 1 - ALTES BLAT
T"
109 POSITION 1,4:? #6;"      LADEN"
120 POSITION 1,6:? #6;" 2 - BLATT NEU"
130 POSITION 1,7:? #6;"      ANLEGEN"
140 ? :? "                    AUSNAHME"
150 OPEN #1,4,0,"K:"
160 GET #1,Z
170 CLOSE #1
180 IF Z=49 THEN 4200
190 IF Z=50 THEN 220
195 GOTO 150
220 DIM N$(12),K$(1),F1$(10)
230 DIM I$(2),FILE$(13),HILF$(9),W$(10
),FEHLP$(12),DA$(8)
240 DIM SPNAM$(189):SPNAM$(1)=" ":SPNA
M$(189)=SPNAM$(1):SPNAM$(2)=SPNAM$(1)
250 DIM FLAG$(252):FLAG$(1)=" ":FLAG
$(252)=FLAG$(1):FLAG$(2)=FLAG$(1)
300 SETCOLOR 2,8,0
305 GRAPHICS 2
307 POSITION 2,3:? #6;" ARBEITSBLATT"
310 POSITION 2,6:? #6;"  DEFINIEREN  "
315 POKE 708,204
320 ? :? "  BITTE IRGENDEINE TASTE DRÜ
CKEN"
330 OPEN #1,4,0,"K:"
331 GET #1,T
335 CLOSE #1
340 GRAPHICS 0:? CHR$(125)
350 TRAP 350
360 POSITION 27,2:? "    ":POSITION 0,
2:? "[illegible]";:INPU
T N1:IF (N1<1)+(N1>96) THEN 360
365 TRAP 365
370 POSITION 27,4:? "    ":POSITION 0,
4:? "[illegible]";:INPU
T M:IF (M<1)+(M>21) THEN 370
375 TRAP 375
380 POSITION 27,6:? "    ":POSITION 0,6
:? "[illegible]";:INPUT
 C1:IF (C1<1)+(C1>21)+(C1>M) THEN 380
385 TRAP 385
390 POSITION 0,12:? "[illegible]
[illegible]";:INPUT DA$:IF DA$="" THEN 3
90
391 IF LEN(DA$)>8 THEN 390
405 D=N1*T:E=M
410 DIM C(N),F$(N),H$(6),D(N,N)
412 FOR Y=1 TO N
413 FOR Z=1 TO M
414 D(Y,Z)=0
416 NEXT Z
418 NEXT Y
420 ? CHR$(125)
425 I$="":I=0:BR=40
440 FOR V=1 TO C1
470 C(V)=(INT(BR/C1)-INT(37/C1)+1)
480 NEXT V
490 IF I$="SV" THEN 1740
500 J=0:R=0:S=0
510 IF I$="A" THEN 1700
512 L=M
520 S=0:IF L>17 THEN L=17
530 FOR T=1 TO L:POSITION 0,T+3:S=S+1:
? CHR$(29);S:NEXT T
570 POKE 752,0:POSITION 0,0:? "[illegible]
[illegible] ";M;" ZEILEN ";" ";N;" SPALTEN  [illegible]
ILFE"
580 POSITION 0,1:FOR X=1 TO 39:? "=";:
NEXT X
590 FOR C=1 TO C1:POSITION (C*INT(BR/C
1))-INT(21/C1),1:? "S";C-1:NEXT C
600 IF I$="LA" THEN GOSUB 1790
610 GOSUB 1070:POSITION 0,22:? "[illegible]
[illegible]";:INPUT I$
```

```
620 IF I$="" THEN 610
624 IF I$="MC" THEN 12000
630 GOSUB 1630
634 IF ((I$="A")*(B1(10))+((I$="E")*(B
1(10)) THEN 570
640 IF I$="E" THEN 1730:REM ANFANG
650 IF I$="D" THEN 540:REM ENDE
660 IF I$="SL" THEN 1720:REM SPALTE LI
NKS
670 IF I$="SR" THEN 1710:REM SPALTE RE
CHTS
680 IF I$="SH" THEN 1940:REM SHIFT
690 IF I$="Z" THEN 2620:REM ZEILE ADDI
EREN
695 IF I$="H" THEN 5000:REM HELP
700 IF I$="SA" THEN 2930:REM ADDIERE
SPALTE
710 IF I$="FE" THEN 1970:REM FORMEL EI
NGEBEN
720 IF I$="RR" THEN 2120:REM RUECKRECH
NEN
730 IF I$="S" THEN 1260:REM SPALTENSUM
ME
740 IF I$="W" THEN 880:REM WERTEINGABE
750 IF I$="N" THEN 1050:REM SPALTENNAM
E
760 IF I$="SB" THEN 1360:REM AENDERN S
PALTENBREITE
770 IF I$="AW" THEN 1410:REM AENDERN W
ERT
780 IF I$="LF" THEN 2830:REM LOESCHE D
ATENFELDER
790 IF I$="L" THEN ? CHR$(125):RUN :RE
M LOESCHE DATEN U. FORMELN
810 IF I$="SP" THEN 3240:REM SPEICHERN
820 IF I$="LO" THEN 4240:REM LADEN
830 IF I$="U" THEN ? CHR$(125):REM :EN
D
840 GOTO 570
880 GOSUB 1600
890 POSITION 0,0:? "WERTEINGABE ( MAX
9 STELLEN )"
900 V=C+I:FLAG$(12*C-11,12*C)="ZAHL
"
910 FOR R=1 TO RZ
920 R1=R-K:IF R1<1 THEN 940
930 POSITION C(V),R1+2:? "W"
935 TRAP 935
940 GOSUB 1070:POSITION 0,22:? "EINGABE
?";:INPUT W$
945 IF LEN(W$)=0 THEN 940
950 IF LEN(W$)>9 THEN 940
960 A(R,C)=VAL(W$):A(R,C)=INT(A(R,C)*1
00+0.5)/100:IF R1<1 THEN 1000
970 GOSUB 1660
980 IF (R1>18)*(V<R1) THEN POSITION 0,
20:? "
      ":GOSUB 2020
990 POSITION 0,21:? "
                    ";
1000 NEXT R
1010 IF R1>16 THEN 1700
1012 IF A(N,C)<>0 THEN GOTO 1280
1015 GOTO 570
1050 GOSUB 1600:REM SPALTENNAME
1060 V=C+I
1070 POSITION 0,0:? "SPALTENNAME ( MAX
9 ZEICHEN )"
1080 GOSUB 1070:SPNAM$(9*C-8,9*C)="
  "
1082 POSITION 0,22:? "EINGABE";:INPUT
HILF$:SPNAM$(9*C-8,9*C)=HILF$
1090 IF SPNAM$(9*C-8,9*C)="         "
THEN 1080
1110 IF (V<1)+(V>C1) THEN 570
1120 POSITION 0,2:? "  ":POSITION C(V)
,2:? "         ":POSITION C(V),2:? SPN
AM$(9*C-8,9*C)
1130 GOTO 570
1160 POSITION 0,2:? "
                 "
1165 FOR C=1-I TO C1-I
1170 IF C>N THEN RETURN
1180 V=C+I
1200 POSITION C(V),2:? SPNAM$(9*C-8,9*
C)
1210 NEXT C
1220 RETURN
```

```
1260 GOSUB 1600:REM SPALTENSUMME
1270 V=C+I
1280 A(N,C)=0
1290 FOR R=1 TO N1:A(N,C)=A(N,C)+A(R,C
):A(N,C)=INT(A(N,C)*100+0.5)/100:NEXT
R
1300 IF (V<1)+(V>C1) THEN 1320
1310 POSITION C(V),20:? "         ":P
OSITION C(V),20:? A(N,C)
1320 IF I$="WS" THEN 2210
1325 GOTO 570
1360 POSITION 0,0:? "SPALTEN AUF [illegible]
( MAX 3)"
1370 GOSUB 1610:IF C>I THEN 1380
1372 C1=C
1375 GOTO 380
1410 R0$="J":REM AENDERN WERT
1420 GOSUB 1600
1430 FHILF$=FLAG$(12*C-11,12*C)
1431 POKE 756,224
1432 IF FHILF$<>"ZAHL        " THEN PO
SITION 0,0:? "FORMELSPALTE AENDERN (J/
N)":GOTO 1460
1440 GOTO 1470
1450 GOSUB 1070:POSITION 0,22:? "EINGA
BE";:INPUT X$
1460 IF X$<>"J" THEN 570
1470 POSITION 0,0:? "ZEILENNUMMER (ZAH
L 000 1 BIS ";R1;")"
1475 TRAP 8000
1480 GOSUB 1070:POSITION 0,22:? "EINGA
BE";:INPUT R
1490 IF (R<1)+(R>R1) THEN 1480
1500 R1=R-K:V=C+I
1510 POSITION 0,0:? "NEUER WERT
                   "
1515 TRAP 8000
1520 GOSUB 1070:POSITION 0,22:? "EINGA
BE";:INPUT W1:A(R,C)=W1
1530 FLAG$(12*C-11,12*C)="ZAHL
"
1540 IF (V<1)+(V>C1)+(R1<1)+(R1>17) TH
EN 1700
1550 GOSUB 1660
1560 IF A(N,C)<>0 THEN 1280
1570 GOTO 570
1600 GOSUB 1630:POSITION 0,0:? "SPALTEN
NUMMER (ZAHL 000 1 BIS ";N;") "
1610 TRAP 8000:GOSUB 1070:POSITION 0,2
2:? "EINGABE";:INPUT C
1620 IF (C<1)+(C>N) THEN 1610
1630 POSITION 0,0:? "
                      ":RETURN
1640 POSITION 0,22:? "B I T T E  W A H
 L E N":RETURN
1660 POSITION C(V),R1+2:? "         ":
POSITION C(V),R1+2:? INT(A(R,C)*100+0.
5)/100:RETURN
1670 POSITION 0,22:? "
                    "
1674 RETURN
1710 IF (C1-I)>=N THEN 570
1711 I=I+1
1714 GOTO 1740
1720 IF I<=0 THEN 570
1725 I=I-1:GOTO 1740
1730 K=R1-17:S=R1
1740 ? CHR$(125)
1750 FOR C=1+I TO C1-I:REM SPALTEN ANF
 SCHIEB
1760 IF C>N THEN 1800
1770 V=C+1
1780 IF SPNAM$(9*C-8,9*C)<>"         "
 THEN POSITION C(V),2:? SPNAM$(9*C-8,9
*C):GOTO 1800
1790 GOTO 1890
1800 FOR R=ABS(K)+1 TO ABS(K)+L
1810 R1=R-K
1820 POSITION C(V),R1+2:IF A(R,C)<>0 T
HEN PRINT INT(A(R,C)*100+0.5)/100
1830 NEXT R
1840 IF A(N,C)<>0 THEN POSITION C(V),2
0:? A(N,C)
1850 NEXT C
1860 IF I$="A" THEN 520
1862 IF I$="LA" THEN 820
1868 S=S-L:GOTO 830
```

```
1960 POSITION 0,0:? "NUMMER DER EINGAB
[illegible]
1906 REM TRAP 1960
1910 GOSUB 1070 POSITION 0,20:? "EINGA
BE";:INPUT E$
1920 IF E$(1)+(E$-1)M-C1) THEN 1910
1930 I=E E$:GOTO 1740
1970 REM TRAP 0200:GOSUB 1660
1975 GOSUB 1660
1980 POSITION 0,0:? "[illegible]
"
1988 FLANG$(12*C-11,12*C)="
"
1990 GOSUB 1070:POSITION 0,21:? "EINGA
BE";:INPUT FNILF$:IF FNILF$="" THEN 19
80
2000 FLANG$(12*C-11,12*C)=FNILF$
2010 V=C+1
2020 FOR R=1 TO R1
2030 RI=R-X
2040 GOSUB 2260
2050 IF (V(1)+(V)C1)+(R1(1)+(R1)>17) TH
EN 2070
2060 GOSUB 1660
2070 NEXT R
2080 IF B(R,C)<>0 THEN 1200
2085 GOTO 570
2120 FOR C=1 TO M
2125 FNILF$=FLANG$(12*C-11,12*C)
2120 IF FNILF$(1,1)="   " THEN 2210
2130 IF FLANG$(12*C-11,12*C)="EORL
  " THEN 2210
2140 V=C+1
2150 FOR R=1 TO R1
2160 R1=R-X:X=B(R,C):GOSUB 2260
2170 IF (V(1)+(V)C1)+(R1(1)+(R1)>17) TH
EN 2190
2180 IF X<>B(R,C) THEN GOSUB 1660
2190 NEXT R
2200 IF B(R,C)<>0 THEN 1200
2210 NEXT C
2220 GOTO 570
2260 B$=FLANG$(12*C-11,12*C)
2261 LAENGE=0:FOR I=1 TO 12
2262 IF B$(I,I)=" " THEN 2264
2263 LAENGE=LAENGE+1
2264 NEXT I
2252 TRAP 2280
2270 IF (VAL(FNILF$)) THEN B(R,C)=VAL(
FNILF$):RETURN
2280 TRAP 0200
2285 IF (B$(1,1)="S")*(LAENGE<4) THEN
B(R,C)=B(R,VAL(B$(2,LAENGE))):RETURN
2290 IF B$(LAENGE,LAENGE)="+" THEN G=1
:GOTO 2360
2300 IF B$(LAENGE,LAENGE)="-" THEN G=2
:GOTO 2360
2310 IF B$(LAENGE,LAENGE)="/" THEN G=3
:GOTO 2360
2320 IF B$(LAENGE,LAENGE)="-" THEN G=4
:GOTO 2360
2330 IF B$(LAENGE,LAENGE)="^" THEN G=5
:GOTO 2360
2335 IF B$(LAENGE,LAENGE)="%" THEN G=6
:GOTO 2360
2340 IF B$(LAENGE,LAENGE)="W" THEN G=7
:GOTO 2360
2350 IF B$(1,2)="ZS" THEN 2370
2360 IF B$(1,1)="S" THEN ZN=0:GOTO 238
0
2370 ZN=0
2380 B$=B$(2,LAENGE-2)
2390 F1=VAL(B$):F1$=STR$(F1)
2395 IF G=7 THEN 2520
2399 LAENGE=LEN(B$)
2400 B$=B$(LAENGE-(LAENGE-LEN(F1$)-2)
),LAENGE)
2410 IF B$(1,1)="Z" THEN ZN=0:GOTO 244
0
2420 IF B$(1,1)="S" THEN ZN=0:GOTO 244
0
2430 F2=VAL(B$)
2435 ON G GOTO 2460,2460,2500,2800,252
6
2260 LAENGE=LEN(B$)
2860 F2=VAL(B$(LAENGE-(LAENGE-2),LAENG
E))
2450 ON G GOTO 2470,2490,2510,2510,252
0,2560
2460 B(R,C)=B(ZN,F1)*F2:B(R,C)=INT(B(R
,C)*100+0.5)/100:RETURN
2470 B(R,C)=B(ZN,F1)*B(ZN,F2):B(R,C)=I
NT(B(R,C)*100+0.5)/100:RETURN
2480 B(R,C)=B(ZN,F1)+F2:RETURN
2490 B(R,C)=B(ZN,F1)+B(ZN,F2):RETURN
2500 B(R,C)=B(ZN,F1)/F2:B(R,C)=INT(B(R
,C)*100+0.5)/100:RETURN
2510 B(R,C)=B(ZN,F1)/B(ZN,F2):B(R,C)=I
NT(B(R,C)*100+0.5)/100:RETURN
2520 B(R,C)=B(ZN,F1)-F2:RETURN
2525 B(R,C)=INT(SQR(B(ZN,F1))*100+0.5)
/100:RETURN
2526 B(R,C)=B(ZN,F1)^F2:B(R,C)=INT(B(R
,C)*100+0.5)/100:RETURN
2528 B(R,C)=B(ZN,F1)^B(ZN,F2):B(R,C)=I
NT(B(R,C)*100+0.5)/100:RETURN
2530 B(R,C)=B(ZN,F1)-B(ZN,F2):RETURN
2560 B(R,C)=B(ZN,F2)/B(ZN,F1)*100-100:
RETURN
2570 B(R,C)=0
2580 LAENGE1=0:FOR I=1 TO 12
2581 IF B$(I,I)<>" " THEN LAENGE1=LAEN
GE1+1
2582 NEXT I
2585 ZN=VAL(B$(3,LAENGE1))
2586 FOR I=ZN TO C-1:B(R,C)=B(R,C)+B(R
,I):NEXT I:RETURN
2620 IF B1+2>0 THEN RETURN
2630 E=E+1:S=S-3+I
2640 POSITION 0,20:? S
2660 POKE 84,21:? :RETURN
2690 GOSUB 1670:J=J+1
2700 IF (S=B1)+(R=B1) THEN 570
2710 R=17+J:R=R+1:S=S+1
2720 POSITION 0,20:? S,:? "
                      ":POSITION 10,
21:? " ":GOSUB 2660
2730 FOR C=1 TO CL-7
2740 IF FLANG$(12*C-11,12*C)="
  " THEN 2760
2750 V=C+1
2760 POSITION C(V),1:IF B(R,C)=0 THEN
 2770
2760 ? B(R,C)
2770 IF B(R,C)<>0 THEN POSITION C(V),2
0:? B(R,C)
2780 NEXT C
2790 GOTO 810
2830 ? CHR$(125)
2840 GOSUB 1650
2850 FOR C=1 TO M
2860 FOR R=1 TO B1:B(R,C)=0:NEXT R
2870 IF B(R,C)<>0 THEN B(R,C)=0
2880 NEXT C
2890 ? CHR$(125):GOTO 560
2930 GOSUB 1660
2900 ? CHR$(125)
2945 GOSUB 1660
2950 FL=0
2960 FOR R=1 TO B1-1
2970 IF B(R,C)<=B(R+1,C) THEN 2990
2980 TEMP=B(R,C):B(R,C)=B(R+1,C):B(R+1
,C)=TEMP:FL=1
2990 NEXT R
3000 IF FL<>0 THEN 2950
3005 GOTO 1740
3240 ? CHR$(125)
3242 FILE$="":FILE$(1,2)="D:"
3246 FILE$(3) 603
3250 OPEN #1,8,0,FILE$
3260 ? #1,FLANG$
3270 ? #1;SPRUNG
3271 ? #1;B1
3272 ? #1;R
3273 ? #1;M
3274 ? #1;C1
3280 FOR I=1 TO M
3290 FOR J=1 TO R
3300 ? #1;B(J,I)
3310 NEXT J
3320 NEXT I
3330 CLOSE #1:GOTO 1740
4240 GRAPHICS 0:CLR
4242 DIM B$(12),N$(1),F1$(10)
```

```
4244 DIM I$(2),FILE$(11),HILF$(9),W$(1
9),FHILF$(12),AN$(8)
4246 DIM SPNAM$(189),SPNAM$(1)=" ":SPN
AM$(189)=SPNAM$(1):SPNAM$(2)=SPNAM$(1)
4248 DIM FLNAM$(252),FLNAM$(1)=" ":FLN
AM$(252)=FLNAM$(1):FLNAM$(2)=FLNAM$(1)
4252 POSITION 3,10:? "DATEINAME";:INPU
T AN$:IF AN$="" THEN 4252
4253 FILE$="":FILE$(1,2)="D:":FILE$(3)
=AN$
4254 TRAP 9100
4259 OPEN #1,4,0,FILE$
4260 INPUT #1;FLNAM$
4270 INPUT #1;SPNAM$
4271 INPUT #1;N1
4272 INPUT #1;N
4273 INPUT #1;M
4274 INPUT #1;C1
4276 L=N1:C=C1*80/40
4278 DIM C(N),FS(M),W$(8),O(N,M)
4279 FOR Y=1 TO N
4280 FOR Z=1 TO M
4281 O(Y,Z)=0
4282 NEXT Z
4283 NEXT Y
4288 FOR I=1 TO M
4290 FOR J=1 TO N
4300 INPUT #1;HILF:O(J,I)=HILF
4310 NEXT J
4320 NEXT I
4330 R=N1*2:L=17:IS="LA":J=0
4331 IF N1<10 THEN L=N1
4332 FOR V=1 TO C1
4334 C(V)=(INT(80/C1)-INT(17/C1)+1)
4338 NEXT V
4340 CLOSE #1:GOTO 1740
5000 ? CHR$(125):POKE 766,208:POKE 82
,0:POKE 752,1
5008 SETCOLOR 2,0,0
5010 POSITION 0,0:? "
     SEITE 1 VON 6"
5015 ? "______________________________
__________"
5030 ? "FOLGENDE KOMMANDOS SIND MÖGLIC
H"
5040 ? :? "   - Fenster um 1 Zeile abw
ärts"
5050 ? :? "   - Fenster zum Blattanfan
g"
5055 ? :? "   - Fenster zum Blattende"
5060 ? :? "   - Fenster eine Spalte l
inks"
5062 ? :? "   - Fenster eine Spalte r
echts"
5064 ? :? "   - Fenster um beliebige
Spalten          schieben"
5066 ? :? "   - Spaltenanzahl am Schi
rm ändern"
5070 ? :? "  - Spaltenname eingeben"
5080 POSITION 0,22:? " RETURN=MEHR HE
LP      ESC=ZUM PROGRAMM"
5086 OPEN #3,4,0,"K:"
5087 GET #3,X
5088 CLOSE #3
5090 IF X=155 THEN 5100
5092 IF X=27 THEN GRAPHICS 0:GOTO 1740
5095 GOTO 5086
5100 ? CHR$(125):POKE 766,208:POKE 82,
0:POKE 752,1
5110 POSITION 0,0:? "
     SEITE 2 VON 6"
5115 ? "______________________________
__________"
5160 ? "   - Errechne Spaltensumme"
5155 ? :? "   - Sortiere eine Spalte"
5158 ? :? "   - Feldweise Werteingabe"
5162 ? :? "   - Wert eines Feldes "än
dern"
5163 ? :? "   - Löschen Felder (Forme
ln bleiben)"
5166 ? :? "  - Anfertigung CALC BOOKL
"
5170 ? :? "  - Arbeitsblatt laden"
5175 ? :? "  - Arbeitsblatt speichern
"
5180 ? :? "  - Hardcopy anfertigen"
5182 ? :? "   - Neues Arbeitsblatt anl
egen"
```

```
5188 POSITION 0,22:? " RETURN=MEHR HE
LP      ESC=ZUM PROGRAMM"
5190 OPEN #3,4,0,"K:"
5195 GET #3,X
5200 CLOSE #3
5210 IF X=155 THEN 5300
5212 IF X=27 THEN GRAPHICS 0:GOTO 1740
5220 GOTO 5190
5300 ? CHR$(125):POKE 766,208:POKE 82,
0:POKE 752,1
5310 POSITION 0,0:? "
     SEITE 3 VON 6"
5315 ? "______________________________
__________"
5330 ? "      Rückrechnung, d.h. alle F
eldinhal-"
5332 ? "      te aller Spalten werden
nach For-"
5334 ? "      mel-od. Wertänderung akt
ualisiert."
5350 ? "   - Formel eingeben"
5355 ? :? "Das berechnete Resultat wir
d in die an-"
5360 ? "gegebene Spalte eingetragen. D
ie Formel"
5363 ? "besteht aus einem oder zwei Op
eranden."
5365 ? "Bei einem Operanden gibt es 3
Möglich "
5370 ? "keiten:"
5380 ? :? " a) eine Zahl (sie wird in
jede Zeile "
5383 ? "      der Spalte eingetragen),"
5390 ? :? " b) S und Spaltennummer,etw
a S1-trägt"
5393 ? "      die Summe der ersten Spalt
e in jede"
5395 ? "      Zeile auf angegebener Spal
te ein."
5400 POSITION 0,22:? " RETURN=MEHR HE
LP      ESC=ZUM PROGRAMM"
5410 OPEN #3,4,0,"K:"
5420 GET #3,X
5430 CLOSE #3
5440 IF X=155 THEN 5500
5450 IF X=27 THEN GRAPHICS 0:GOTO 1740
5460 GOTO 5410
5500 ? CHR$(125):POKE 766,208:POKE 82,
0:POKE 752,1
5510 POSITION 0,0:? "
     SEITE 4 VON 6"
5515 ? "______________________________
__________"
5520 ? " c) ZS Spaltennummer, etwa ZS2
   bildet"
5522 ? "      die Zeilensummen ab der Sp
alte 2"
5550 ? "Mit zwei Operanden gilt das Fo
rmat:"
5555 ? :? "     OPERAND1,OPERAND2:OPERA
TOR"
5560 ? :? "Der erste Operand darf kein
e Zahl sein."
5565 ? "Er muss K(Spaltennummer),etwa
K3 "
5567 ? "oder S(Spaltennummer), etwa S2
 lauten. "
5570 ? "Der zweite Operand kann eine Z
ahl oder "
5575 ? "K(Spaltennummer) oder S(Spalte
nnummer )"
5580 ? "sein. Als letztes folgt der Op
erator "
5585 ? :? "         (Grundrechnungsart
en)"
5590 ? :? "     (Wurzelziehen)   (Pot
enzieren)"
5595 ? :? "   proz. Differenz zwischen
Op 1 u.Op 2"
5600 POSITION 0,22:? " RETURN=MEHR HE
LP      ESC=ZUM PROGRAMM"
5610 OPEN #3,4,0,"K:"
5615 GET #3,X
5620 CLOSE #3
5625 IF X=155 THEN 5700
5630 IF X=27 THEN GRAPHICS 0:GOTO 1740
5700 ? CHR$(125):POKE 766,208:POKE 82,
0:POKE 752,1
```

IRATA

MIERENDORF
PLATZ 8
1 BERLIN 10
030/3453061

VIDEOMEISTER

DER VIDEOMEISTER IST EINE ENTWICKLUNG UM BILDER VON CAMERA ODER VIDEO IN DEN ATARI ZU LESEN. MIT DEM PROGRAMM KANN MAN GRAUWERTE AENDERN UND AUCH FARBBILDER ERSTELLEN. DIE SOFTWARE HAT PROFISTANDARD

199.-

DRUCKINTERFACE

EINE VERBINDUNG VON COMPUTER ZUM DRUCKER OHNE VORHER SCHALTER ZU STELLEN ODER EIN PROGRAMM ZU LADEN. ES WIRD NUR IN DIE FLOPPY GESTECKT UND MIT DEM DRUCKER VERBUNDEN.

199.-

SUPERLABEL 2

BILDER UND SCHRIFTEN AUF ETIKETTEN BRINGEN WAR SCHON IMMER EIN TRAUM. ETIKETTEN IN DREI GROESSEN KANN MAN BENUTZEN. DER DRUCKER SOLLTE EPSON KOMP. SEIN

40.-

512K RAM

399.-

ENDLICH EINMAL RICHTIG SPEICHER HABEN. DAS WAR DER WUNSCH. JETZT KANN MAN DAS HABEN. DIE 512K KARTE IST LEICHT IN DEN XL ZU INSTALLIEREN

DAS HSB WIRD IN DIE 1050 EINGEBAUT. DAS LAUFWERK HAT NUN 172K UND WIRD ERHEBLICH SCHNELLER. VON FAST JEDER DISK IST EINE COPY ZU ERSTELLEN. DER EINBAU IST EINFACH UND OHNE LOETEN

HIGH SPEED 1050

149.-

SOUND MEISTER

DER SOUNDMEISTER IST EIN INTERFACE MIT DEM MAN MUSIK ODER SPRACHE VOM RADIO ODER MICROFON IN DEN COMPUTER SPEICHERT. AUCH HIER GIBT ES ECHTE PROFISOFTWARE DAZU. MIT DER 512K KARTE IST PLATZ FUER 3 MINUTEN.

129.-

EDITOR

BILDER UND SCHRIFTEN KANN MAN LEICHT MIT DEM PROGRAMM VERARBEITEN. GUT GEEIGNET FUER KLEINE ZEITUNGEN ODER INSERATE. DAS GANZE PROGRAMM BESTEHT AUS 10 DISKETTEN

149.-

SEIKOSHA	HARDCOPY
GP 100	30.-
GP 100 GT	30.-
GP 500 GT	30.-
GP 500	30.-
GP 55R AT	30.-

PRINT SHOP DRUCK	
GP 100	20.-
GP 500	30.-

HARDCOPY	
ATARI 1020	20.-
EPSON	20.-

PROGRAMME	
VOKABEL	20.-
DATEI 7.0	30.-
CATLOG	20.-
PROTECT	20.-
UNDELETER	20.-
SUPERCOPY	30.-
KATALOG-INFO	00.-

SPEED MEISTER

DOS UND DUP LADEN IN 5 SEKUNDEN
SECTORCOPY 1 DISK 17 SEKUNDEN
SECTORCOPY DD 25 SEKUNDEN
62 SECTOREN BILD 4 SEKUNDEN
EPROMBRENNER ANSCHLIESSBAR
CENTRONICS INTERFACE AUSGANG
24 BIT USER PORT
HIGH SPEED BOARD ERFORDERLICH

COMPUTERCLUB
SEIT 1981
2700 LEUTE
EUROPAWEIT
INFO FREI
030 343 30 81

PANASONIC DRUCKER	
KXP 1080	690.-
KXP 1091	781.-
KXP 1092	1050.-

FARBBAENDER	
SCHWARZ	24.-
ROT	24.-
BLAU	24.-
GRUEN	24.-
GRAFIKZEICHEN	10.-

BUECHER	
EINSTEIGER	15.-
DISK BUCH	15.-
C/65 COMPILER	30.-
FILE FAX	30.-
TELFON TECADO	35.-

MASCHINE LERNEN MIT EINEM SPIEL. ALLE ROUTINEN SICHTBAR.

IRATA MIERENDORFPLATZ 8 BERLIN 10

```
5710 POSITION 0,0:? "              [illegible]
[illegible]    SEITE 5 VON 6"
5715 ? "________________________________
________"
5720 ? "Zu trennen sind die Operanden
und der"
5740 ? "Operator durch Strichpunkt (Do
r Doppel-"
5750 ? "punkt oder Komma."
5760 ? "                      BEISPIELE"
5771 ? :? "X1;X1,0   multipliziert die
 Felder der"
5772 ? "   Spalte1 mit den Feldern der
 Spalte5"
5774 ? "   und tr gt die Ergebnisse in
 die an-"
5776 ? "   gegebene Spalte ein."
5776 ? :? "X4,S2;"   addiert zu jedem
Feld der "
5777 ? "   Spalte4 die Spaltensumme de
r Spalte2"
5778 ? "   und tr gt die Ergebnisse in
 die an-"
5779 ? "   gegebene Spalte ein."
5800 ? :? "X5;M - berechnet die derzei
n der Fel-"
5802 ? "   der der Spalte5 und tr gt d
ie Ergeb-"
5804 ? "   nisse in die angegebene Spa
lte ein."
6000 POSITION 0,22:? "  [illegible]=MEHR HE
LP  [illegible]=ZUM PROGRAMM"
6010 OPEN #1,4,0,"K:"
6015 GET #1,X
6020 CLOSE #1
6030 IF X=160 THEN 6570
6035 IF X=27 THEN GRAPHICS 0:GOTO 1740
6040 GOTO 6010
6070 ? CHR$(125):POKE 750,204:POKE 82,
0 POKE 752,1
6075 POSITION 0,0:? "              [illegible]
[illegible]    SEITE 6 VON 6"
6080 ? "________________________________
________"
6088 ? "%10;2.5;/ - dividiert jedes Fe
ld der"
6090 ? "   Spalte10 durch den Faktor 2.5
 und "
6092 ? "   tr gt die Ergebnisse in die
 angege-"
6094 ? "   bene Spalte ein "
```

```
6900 ? :? "%5,2,^ - quadriert jedes Fe
ld der"
6905 ? "   Spalte5 und tr gt die Ergeb
nisse  "
6910 ? "   in die angegebene Spalte ei
n."
6911 ? :? "================================
========"
6913 ? " Zum Kennenlernen des Programm
s kann-"
6914 ? " nen Sie das Arbeitsblatt DEMO
 laden"
6915 ? "================================
========"
6918 ? "  A C H T U N G ! ! "
6920 ? :? " Das Programm rundet alle E
ingaben und"
6925 ? " Berechnungen auf 2 Kommastell
en ab "
6975 POSITION 0,22 ? "  [illegible]=[illegible]
HELP  [illegible]=ZUM PROGRAMM"
6980 OPEN #1,4,0,"K:"
6982 GET #1,X
6984 CLOSE #1
6986 IF X=160 THEN 5000
6988 IF X=27 THEN GRAPHICS 0:GOTO 1740
6990 GOTO 6010
7000 END
8000 ? CHR$(125):POSITION 4,10:? "FALS
CHE EINGABE !!!!";? CHR$(253):? CHR$(2
53):GOTO 1740
8100 POSITION 4,17:? "PLOTT NICHT MOEG
LICH !":FOR Z=1 TO 100:NEXT Z
8150 CLOSE #1:? CHR$(125):GOTO 6251
12000 A0=PEEK(88)+PEEK(89)*256:Z=0
12050 OPEN #2,8,0,"P:"
12100 FOR I=0 TO 959
12110 B=PEEK(A0):A0=A0+1:Z=Z+1
12111 IF B>=0 AND B<=63 THEN B=B+32:GO
TO 12120
12112 IF B>=64 AND B<=95 THEN B=B-64:G
OTO 12120
12113 IF B>=128 AND B<=191 THEN B=B+32
:GOTO 12120
12114 IF B>=192 AND B<=223 THEN B=B-64
:GOTO 12120
12120 PUT #2,B
12130 IF Z>39 THEN Z=0:? #2
12150 NEXT I
12200 CLOSE #2:GOTO 570
```

Hardcopy-Programm für Seikosha GP 500AT

Mit diesem Programm können Sie eine Hardcopy vom GRAPHICS 8 Bildschirm auf dem Seikosha GP 500AT ausgeben. Das Programm ist vollkommen in Maschinensprache geschrieben und daher erfolgt der Ausdruck sehr schnell. Im Listing wird zuerst der Maschinencode in den Bereich von $0600-$06FE eingelesen, dann erscheint ein kleines Demo auf dem Bildschirm. Das Hardcopy Programm wird folgendermaßen aufgerufen:

```
OPEN #KANAL,8,0,"P:"
? #KANAL;CHR$(27);CHR$(57)
X=USR(1536,KANAL*16)
```

Falls während des Ausdrucks ein Fehler auftreten sollte, so steht seine Nummer in der Variablen X.

Noch ein wichtiger Hinweis: Nach meinen Informationen ist der Drucker 1029 von Atari kompatibel zum Seikosha GP 500AT. Somit könnte dieses Programm auch für den 1029 zu verwenden sein.

Matthias Wunder

```
100 REM *** Hardcopy fuer Seikosha
110 REM *** GP-500AT
120 REM ***
130 REM *** (c) Matthias Wunder
140 REM
150 S=0
160 FOR A=1536 TO 1790
170 READ I:S=S+I
180 POKE A,I
190 NEXT A
200 IF S<>30910 THEN ? CHR$(253);"DATE
NFEHLER!!!":END
210 REM
220 REM *** Kleines Demo ***
230 REM
240 GRAPHICS 24:COLOR 1
250 FOR A=0 TO 319 STEP 10
260 PLOT 160,191:DRAWTO A,0
270 NEXT A
280 REM
```

```
290 REM *** Hardcopy ***
300 REM
310 KANAL=1
320 OPEN #KANAL,8,0,"P:"
330 ? #KANAL;CHR$(27);CHR$(27);CHR$(57
):REM Select 9 Lines per Inch
340 X=USR(1536,KANAL*16)
350 IF X>=128 THEN ? "FEHLER ";X
360 END
30000 DATA 104,104,104,141,255,6,169
30001 DATA 0,133,203,160,0,162,3
30002 DATA 189,231,6,32,206,6,203
30003 DATA 16,247,162,7,169,0,157
30004 DATA 0,4,202,16,250,133,204
30005 DATA 189,7,186,203,224,189,208
30006 DATA 1,76,133,207,169,0,133
30007 DATA 203,133,206,162,204,32,169
30008 DATA 6,160,205,32,169,6,24
30009 DATA 160,80,101,205,133,205,165
30010 DATA 99,101,206,133,206,177,205
30011 DATA 162,7,42,72,104,20,138
30012 DATA 72,56,189,6,229,204,170
30013 DATA 189,0,56,42,202,16,252
30014 DATA 133,208,104,170,24,165,208
30015 DATA 125,0,4,157,0,4,104
30016 DATA 202,16,220,230,204,165,204
30017 DATA 197,207,144,160,162,7,189
30018 DATA 0,4,32,206,6,202,16
30019 DATA 247,208,192,80,144,160,165
30020 DATA 207,201,3,240,15,24,169
30021 DATA 7,101,205,133,203,169,130
30022 DATA 32,206,6,76,10,6,169
30023 DATA 1,133,212,169,0,133,213
30024 DATA 96,133,208,138,72,169,0
30025 DATA 133,209,166,208,208,11,24
30026 DATA 105,48,144,2,230,209,202
30027 DATA 76,177,6,24,101,209,133
30028 DATA 205,165,206,101,209,133,206
30029 DATA 104,170,96,133,208,138,72
30030 DATA 152,72,174,255,6,189,11
30031 DATA 157,66,3,169,0,157,72
30032 DATA 3,157,73,3,165,208,32
30033 DATA 86,228,132,208,104,168,104
30034 DATA 170,165,208,16,7,104,104
30035 DATA 165,208,76,162,6,96,64
30036 DATA 1,65,27
```

Speed-Tape für Atari XL/XE mit 64 KByte

Nachdem Sie "Speed-Tape" eingetippt haben, starten Sie es mit RUN. Sollten Datenfehler auftauchen, bessern Sie diese aus und starten Sie das Programm erneut. Wenn das Programm fehlerfrei gelaufen ist, dann drücken Sie RESET und speichern es ab. Der Basic-Teil des Programms kann nun gelöscht werden. "Speed"-Tape benötigt das unter dem OS-ROM liegende RAM. Es bleibt solange im Speicher, bis dieses RAM beschrieben wird (z.B. durch eine RAM-Disk oder wenn man den Computer ausschaltet). Weder NEW noch RESET gefährden "Speed-Tape".

"Speed-Tape" wird durch POKE 54017,253 oder RESET ausgeschaltet und durch POKE 54017,252 aktiviert. Nach diesem POKE-Befehl laufen sämtliche Cassettenoperationen mit höherer Geschwindigkeit ab. Alle Befehle bleiben gleich, und es kommt auch kein neuer dazu.

Programme, die mit "Speed-Tape" gespeichert wurden, lassen sich auch nur damit laden. Will man ein Programm auf das "Speed-Tape"-Format umwandeln, muß man "Speed-Tape" laden und starten, schaltet es dann aus und lädt das betreffende Programm. Danach muß man "Speed-Tape" einschalten und das Programm speichern.

Da "Speed-Tape" mit sehr hoher Baudrate speichert, können leicht Übertragungsfehler auftreten. Deshalb sollte man immer Rückkopien in normaler Geschwindigkeit anfertigen. Es sollte auch nur eingeschaltet sein, wenn man es benötigt. Ansonsten ist es besser, "Speed-Tape" auszuschalten, da sonst das Betriebssystem (das ins RAM kopiert wurde, damit das System nicht abstürzt) ungeschützt ist und bei unkontrollierten POKE-Befehlen leicht abstürzt. "Speed-Tape" erhöht lediglich die Baudrate von 600 Bit/sec auf 1150 Bit/sec, der Leader und die Pausen zwischen den Blöcken bleiben gleich lang. Daher kann erst bei größeren Datenmengen eine optimale Wirkung erzielt werden (ab ca. 16 KByte).

Andreas Frenger

Speed-Tape 1,0

```
1000 REM ******************
1002 REM *                *
1005 REM *  SPEED-TAPE 1.0  *
1010 REM *  (c) by FAB 1986 *
1020 REM *  fuer alle XL,XE *
1030 REM *                *
1040 REM ******************
1050 SETCOLOR 2,0,2:SETCOLOR 4,0,4:SET
COLOR 1,0:POKE 752,1
1070 ? "}":? :? " SPEED-TAPE 1.0 "
1080 ? :? "POKE 54017,253 : SPEED-T
APE EIN":? "POKE 54017,252"
1090 ? "oder RESET : SPEED-TAPE AUS"
1100 GOSUB 10000:IF PEEK(2948):POKE 540
17,252
1110 ? :? :? :? "SPEED-TAPE ist aktivi
ert."
1120 POKE 60323,200:POKE 60320,2:A=6078
04:POKE A,169:POKE A+1,250:POKE A+2,23
4
1130 POKE A+7,169:POKE A+8,2:POKE A+9,
234
1140 A=60816:POKE A,32:POKE A,237
1150 POKE 752,0:END
10000 REM ---ROM INS RAM KOPIEREN---
10010 CHK=0:RESTORE 10100
10020 FOR I=24000 TO 24592:READ A
10030 IF D>255 THEN IF CHK<>D THEN ? "
DATENFEHLER IN ZEILE";ABCD:END
10035 IF D>255 THEN CHK=0:LINE=LINE+10
10040 CHK=CHK+D:POKE I,D:NEXT I
10052 IF D<>0 THEN 10030
10050 RETURN
10060 ? PEEK(183)+PEEK(184)*256:END
10100 DATA 104,120,169,0,141,14,212,16
9,202,141,1,211,169,192,2099
10110 DATA 133,204,169,207,133,206,169
,0,133,203,133,205,160,3055
10120 DATA 230,169,253,141,1,211,177,2
05,72,169,252,141,1,211,3256
10130 DATA 238,104,145,205,136,208,233
,230,204,230,206,165,206,197,3478
10140 DATA 234,206,208,225,169,216,133
,204,169,81,133,206,169,1,133,3158
10150 DATA 203,160,8,133,205,169,197,1
17,29,161,4,193,38,172,3223
10160 DATA 216,205,169,153,133,208,169
,241,161,227,133,206,169,1,3121
10170 DATA 213,153,141,1,211,169,64,14
1,14,212,88,104,2899
10180 DATA 212,88,104,255,3458
```

Filecopy

Mit diesem Programm lassen sich relativ leicht Files kopieren. Es läuft auf allen 8-Bit Atari-Computern mit mindestens 16 KByte RAM und einer Diskettenstation und hat folgende Vorzüge:

1. Arbeitet unter jeder DOS-Version
2. Kopiert maximal 64 Programme (eine Directory)
3. Kein Eintippen der Filenamen
4. Pro Durchgang werden ca. 300 Sektoren geladen (bei 64 KByte RAM)
5. Es werden mehrere Files in einem Durchgang kopiert
6. Freie Wahl der Laufwerke
7. Alle Fehler werden abgefangen
8. Nach dem Kopiervorgang werden alle Files auf der Zieldiskette gesichert
9. "Filecopy" ist nur 15 Sektoren lang (weniger als 2 KByte)

Norbert Hagemann

Filecopy

```
16 REM *     (c) Feb.1986 by
17 REM *     Norbert Hagemann        *
100 DIM D$(95):OPEN #1,8,0,"D:AUTORUN.
SYS"
110 ? "OBJ-File wird generiert....":?
:TRAP 190:S=0
120 READ D$:READ P
130 ? "*";
140 FOR I=1 TO LEN(D$) STEP 2
150 H=ASC(D$(I))-48:L=ASC(D$(I+1))-48
160 D=(H-(H>9)*7)*16+L-(L>9)*7:S=S+D:P
UT #1,D:IF S>999 THEN S=S-1000
170 NEXT I:IF S=P THEN 120
180 ? :? :? "Datenfehler in Zeile ";PE
EK(183)+PEEK(184)*256:CLOSE #1:STOP
190 IF PEEK(195)<>6 THEN 180
200 CLOSE #1
210 ? :? :? "OBJ-File ordnungsgemaess
erzeugt."
220 END
1000 DATA FFFF0020F02B7B6042232000042FF
FF0202020202070202020202020202020202
0202020202024100200004AE9EC,608
1010 DATA E5E0A1EFF0F9808E30900919996
9600E2F9800EEFF262E5F2F49060E1E7E5EDE1
EEEE80C04015A5102977F0510,250
1020 DATA 8D0ED2A9B0A22000300286310249
3185000509A2FFB6C0E90641204A0509C52DE9
2A69D191950970A0129195A2,810
1030 DATA 1620C326A026A10020F02A69FC91
95A0129195A0279195CA0DE720C326A9C32DE9
2A69DA9195A9D0A0129195A2,727
1040 DATA 1720D026E020E026A200F0402660
0000F920F626200A4229D1C4A0800A6054E120
5F260009710EA4A0BEAA6B0A0,795
1050 DATA 4020F26A6001DCA20E020D026E0
0E0DF9A9040056AE9CC20532520562DA903A006
FC20F721A60020F924A90FA0,475
1060 DATA 2705F3047445F3104A03A5F49D45
03A9149D4005A9009D4903A9059D42032056E4
300720CF20E6A0D073C0B020,776
1070 DATA 6E205525A5F3A4F4B5004D1A27F
A90095000SCA10F4C6A07005A2164DC022200C
2620C425A9FA20F82630642 5,56
1080 DATA 9D344A0007202F250C69E94A0
EE203F252093260006902270950250D0B0010
F5300A6CF0D000549D0D00,42
1090 DATA 0520C4252027250FCFA23F000005
2000EA10F0009240D24200002605CC45002067
24203226A50A0105C106CE,105
1100 DATA 203F250D0005D007205F2590F300
7020C425A6CF00C005D0A2093260210000420
0024A210A4CFA5C090440399,998
1110 DATA 0006A5CE9049039900063A0E502
E5CD9D4003F021F322A0E0020DE50490A9D9
9E42032056E4A6CF0910C005,211
1120 DATA 1021C08ED047001019B9490330D0
0650C005C00945039DC0065CE05E20B3200C
D021A0100940059D0060949,902
1130 DATA 0590C08E6CC0039D067242035026
200C26203F250D0B0D0A202F2590F3000E4C
0124200425805C9100CA6CF20,429
1140 DATA 9526A220A0020302400CFA22009
0006914403094006970450390006930306C0
069D490309002DC72A4CF99,058
1150 DATA C0090CE10A4905990005205625
10FF4CA521A23F0C3005DF6CA10F0A9230000
A6092090242000425A21220A0,340
1160 DATA 264C132053 36F75726365002472 69
76650010101E0011FF246573746 96E0024 7269
7665010101E001FF900F00,065
1170 DATA 0A90FAE0DFA22EF2300236A16E
6769FF336F757262650024 72697665FF336574
0F236CA56172FF010000000,381
1180 DATA 00236F7079666C6167FF0005A50C
A5A3040000236E616E6765FF246573746 96E00
2472697665FF33626C656374,390
1190 DATA FF00000000000026696C656E616E
65FF00B50A102204900024697265637 46F7279
FF64726F600336F75726565,12
1200 DATA 00247269766 5FF626567696E0074
6865FF000000000000000000236F70790000000000
00FF0A6A0A0025 52322F3200,732
1210 DATA 000000000 0A0FF256E6006FA6
00236F70740E0E0E0E0E0E0E0E1EFF0B0C44102BA
000072657374617274FF0000,373
1220 DATA 007466650030726F6772616D60FF
0A00A0A002E6F0026696C657301010000000FF
337469F023E8246C6C00776F,057
1230 DATA 726B696E670E0E0E0E0E0FF296E73
65727400336F7572636500246973 60FF296E73
65727400246573746 96E0026,845
1240 DATA 697360FFA1AE34007072657373 00
000364A18204000 0FFC6220922EC22FC222223
32255236723092313233F23,459
1250 DATA 462379230323592300241324 2624
A7230023CA23EE230A230D0027A500C509702C
307C2A020FA5CCF001E02048,792
1260 DATA 26A21120402 69A0ED01024A0000
0220FA26A300D2800102202725 20A6250F060
8E0027A22A0E02270E0427A2,973
1270 DATA 2E0E0527A21020A9034A9FF9D40
03A9269D45039D4003A9009D4003A9034203
2056E410C6A6A04DA90005,318
1280 DATA 05205325205625A2E20402620A0
0920EA0BA219206DEC24E72526A00201F3291F
919508 10F7A6CC206C26A012,23
1290 DATA A95991950801950490919 5B010F7
CA96A9D99195207C26A2132040262040262082
24202 7254C61204 01F00CA07,669
1300 DATA 00F960E60CA5BCC0000012C91400
0EC60CEA0011EA500650C05CFAAC5000 60A35604
99059500966 0A2107CA22009,037
1310 DATA 0C4CC7244 44F032E535953AC1F00
A9FF20F826CC1F3000F3440C007F0038C1F00
9644600C0CA4CF1C4794 0CA0,429
1320 DATA CF25479C0C5C1C1C442C940C1E3C
0A526A2900000101020202020101102020203
020202030202020502010300,135
1330 DATA A910A5F3A92785F4CA300520CF26
30F0A0020060A46006 05A21A206C2620C326
A6052003 25A6C5A040C05AFA4,015
1340 DATA AF01F330E9201D00E025E32605C0
C0C09195C60F10E30002A9809195810F3A00C
C50A0000002093E26919506,770
1350 DATA 10F0E60A6605A5D4C914F006A5D5
C900A006A000F002A95C0461915 0A2501B0590
6914C5003002A2004401091,085
1360 DATA 954010101E207C26A2150A000000
392405F3095A2405F4206C26A60001F3C9FFF0
059195C800F9E0A4C8E03540,600
1370 DATA 02A2310A291F9150A010707D2565
90057500 9E25659105966 0A20720 6C26A200A0
12A9009195 0E10F030C326CA,176
1380 DATA 00F160208325A001A2001F3C920
F00993 0227C00E000D0F1A92E920227000 91
F3A00327CBE0C0C00F560A0,60
1390 DATA 00050050C0A92010659003 9590
00E6966010A5F369000F3900E6F450206C26
A000A9C19195E4262727A015,290
1400 DATA A9C420C00279195B0A02919586 00
F360A9EE05 14A51400FCA04451 3A050CA600 0E
4C208E40200940A220050C06,70)
1410 DATA 00A201004 9CA0E4402204 4A25000 7
200C0F204C4620E002E10 70227,617
```

Zeichensatzfinder

Neue Zeichen auf dem Atari: um dieses Thema drehten sich in unserem Heft schon mehrere Artikel, so zum Beispiel die beiden bisher veröffentlichten Zeichensatzeditoren. Bei dem Hilfsprogramm unseres Stammlesers Norbert Hagemann geht es jedoch nicht darum, eigene Fonts Zeichen für Zeichen zu basteln, sondern fertige Zeichensätze auszubauen, wie sie in vielen professionellen Programmen enthalten sind.

Als erstes sollten Sie sich eine DOS-II-Masterdisk erstellen, indem Sie eine Diskette formatieren und mit der DOS II Menü Option "H" das DOS darauf schreiben. Tippen Sie dann das Programm fehlerfrei ein und speichern Sie es sicherheitshalber gleich einmal ab. Wenn Sie jetzt die neue Systemdisk einlegen und RUN eingeben, so sollte außer der Meldung "OBJ-File wird generiert..." eine immer länger werdende Reihe aus Sternchen erscheinen. Wird ein fehlerhaftes Datum entdeckt, so wird die entsprechende Zeile reklamiert. Wenn schließlich die O.K.-Meldung das Ende signalisiert, befindet sich das File "AUTORUN.SYS" auf der Diskette. Wenn Sie jetzt den Computer aus- und (möglichst mit gedrückter OPTION-Taste) wieder anschalten, so wird der Zeichensatzfinder automatisch geladen und gestartet. Das Programm meldet sich mit einem schwarzen Bildschirm, auf dem unter der Copyright-Laufschrift ein graues Raster und ein vier Punkte umfassendes Menü erscheinen.

Versuchen wir als erstes Punkt 1: Speicher durchsuchen. Der Computer stellt jetzt den Speicherbereich ab Adresse $E000 im Raster dar. Es handelt sich dabei um den Systemzeichensatz. Mit den Tasten "<" und ">" kann der dargestellte Bereich jetzt um 128 Byte verschoben werden. Die dargestellte Adresse wird dabei angezeigt. Wenn Sie zum Beispiel 40 mal (für 20 Pages) ">" drücken, so erscheint der internationale Zeichensatz, in dem auch die deutschen Umlaute enthalten sind. Als Adresse wird hier $CC00 angegeben. Wenn Sie jetzt A und E000 eingeben, so erscheint wieder der ROM-Font. Sie haben also manuell eine neue Adresse eingegeben. Mit RETURN übergeben Sie die Adreßsteuerung wieder den angegebenen Tasten. Außer diesen beiden Zeichensätzen werden Sie normalerweise wenig Glück bei der Suche haben, es sei denn, Sie hatten zuvor ein Programm im Speicher, dessen Zeichensatz noch vorhanden ist. Dies ist jedoch nur dann möglich, wenn Sie den "Zeichensatzfinder" nicht durch Booten starten, sondern umbenennen (z.B. in FINDER.COM). Sie müssen dann, nachdem Ihr zu untersuchendes Programm unterbrochen wurde, ins DOS gehen und den "Zeichensatzfinder" manuell mit der "L"-Option laden.

Machen Sie sich am besten zuerst mit dem Erscheinungsbild eines Zeichensatzes vertraut. Im obersten linken Kästchen steht das Zeichen, das als Leerzeichen verwendet wird. Ist hier ein anderer Buchstabe abgebildet, so sollten Sie mit "-" und "=" blättern. Sind die Buchstaben über die Kästchen hinweg verschoben, so helfen "+" und "*". Mit ESC kommen Sie wieder zum Hauptmenü zurück.

Versuchen Sie jetzt einmal Funktion 2: Diskette absuchen. Hier fragt das Programm nach der Sektornummer, wobei 0 Default ist. Sollten Sie über eine Bootdiskette verfügen, auf der sich ein Programm mit eigenem Zeichensatz befindet, so können Sie die Diskette mit dieser Option nach Zeichensätzen durchsuchen, wobei die oben erwähnten Tasten nun dazu dienen, um 8, 4 oder 1 Sektor(en) vor- bzw. zurückzublättern. Nach S kann auch eine Sektornummer manuell eingestellt werden. Sollten bei manchen Sektoren Lesefehler irgendeiner Art auftreten, so deutet dies nicht immer auf ein defektes Programm hin, wahrscheinlich haben Sie den Kopierschutz erwischt.

Programme, die sich als File auf Diskette befinden, kann man mit Option 3 untersuchen. Sie werden unter Angabe des Namens in den Speicher geladen und können dort durchsucht werden – genau wie bei Punkt 1. Sollten Sie den Inhalt der Diskette nicht im Kopf haben, so genügt die Eingabe von RETURN, um ein Inhaltsverzeichnis zu bekommen, wobei dies mit einer beliebigen Taste fortgesetzt werden kann. Denken Sie daran, daß Sie bei Basic-Programmen nichts finden können. Die Zeichen sind hier normalerweise in DATA-Zeilen gespeichert, die erst nach RUN in den Speicher gepokt werden.

Ist es Ihnen gelungen, mit einer der drei obigen Optionen einen Zeichensatz auf den Bildschirm zu bekommen, so können Sie ihn nun mit der vierten und letzten Funktion abspeichern. Geben Sie einfach einen Namen ein. Wenn Sie sich erst den Inhalt der Disk ansehen möchten, so können Sie auch hier RETURN verwenden. Der Zeichensatz kann nun von 99% aller Zeichensatzeditoren eingelesen und weiterverarbeitet werden.

Mit dem Hilfsprogramm "Ladedemo" können Sie sich den "entführten" Zeichensatz ansehen. Ersetzen Sie "ZEICHENSATZ" durch den gewünschten Namen und starten Sie mit RUN.

Norbert Hagemann und Thomas Tausend

MC-Generator

```
10 REM
11 REM +------------------------------+
12 REM !                              !
13 REM !      Zeichensatzfinder       !
14 REM !                              !
15 REM !  fuer alle ATARI-Computer    !
16 REM !                              !
17 REM +------------------------------+
18 REM !                              !
19 REM !   (c) June 1986 by           !
20 REM !   Norbert Hagemann           !
21 REM !   Behmstrasse 1              !
22 REM !   1000 Berlin 65             !
23 REM !   Tel.030-493 88 92          !
24 REM !                              !
25 REM +------------------------------+
26 REM !                              !
27 REM ! Mit diesem Programm kann     !
28 REM ! man Zeichensaetze suchen     !
29 REM ! die in anderen Programmen    !
30 REM ! verwendet werden.            !
31 REM ! Wurde ein Zeichensatz ge-    !
32 REM ! funden,so kann dieser auf    !
33 REM ! der Diskette abgespeichert   !
34 REM ! werden.                      !
35 REM ! Das Programm hat folgende    !
36 REM ! Optionen:                    !
37 REM !                              !
38 REM !   [1] Speicher absuchen      !
39 REM !   [2] Diskette absuchen      !
40 REM !   [3] Programm absuchen      !
41 REM !   [4] Datei abspeichern      !
42 REM !                              !
43 REM ! Wird bei der Option 3+4      !
44 REM ! anstelle des Filenamen       !
45 REM ! nur die RETURN-Taste ge-     !
46 REM ! drueckt, wird der Disk-      !
47 REM ! Inhalt angezeigt.            !
48 REM !                              !
49 REM +------------------------------+
50 REM
51 REM
100 DIM D$(95):OPEN #1,8,0,"D:AUTORUN
SYS"
110 ? "OBJ-File wird generiert....":?
:TRAP 190:S=0
120 READ D$:READ P
130 ? "*";
140 FOR I=1 TO LEN(D$) STEP 2
150 H=ASC(D$(I))-48:L=ASC(D$(I+1))-48
160 D=(H-(H>9)*7)*16+L-(L>9)*7:S=S+D:P
UT #1,D:IF S>999 THEN S=S-1000
170 NEXT I:IF S=P THEN 120
180 ? :? :? "Datenfehler in Zeile ";PE
```

```
EK(183)+PEEK(188)*256:CLOSE #1:STOP
190 IF PEEK(1951)<>6 THEN 190
200 CLOSE #1
210 ? :? :? "DOS-File ordnungsgemaess
erzeugt."
220 END
1000 DATA FFFF004F0A04C004570604F005B
527604F0D20300F0F0F0F0F0F0F0F0F0F0F0F
0F0F0F0F0F0F0F0F0F0F0F0F0F,384
1010 DATA 0F0F0F0F0F0F0F0F0F0F0F0F0F0F
0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F
0F0F0F0F0F0F0F0F0F0F0F0F,981
1020 DATA 0F0F0F0F0F0F0F0F0F0F0F0F0F0F
0F0F0F0F0F0F3042C2040202020020202024100
4000FEFF0000E5E9E3E0E0EE,149
1030 DATA F3E1F4FAE6E9E0E4C5F20000E389
0091999960E2F900AEEF2E25F2F4000E1
E7E2E0D01CEEE00FEFF000085,293
1040 DATA E9E3E0E5EEF3E1F4FAE6E9E0E4E5
F20000E3999091999960E2F900E0EEF200
0055555555553555555550,491
1050 DATA 5555555555505555555550055555
550000000000054900555550060555555
FC40F741555555555555555,84
1060 DATA 55555A054F090000000000000059
0091E00E0E03F0E3E9E2E0E5F200E1E2F3F0E3
E0E5E0E590000000000059,32
1070 DATA E90092000E0EA409F3E0E3F4F4E5
00E1E2F3F5E3E0E9E0E59009000000000000
E0936000E0E00F2EFE7F2E1E0,880
1080 DATA E0000E1E2F3F5E3E0E5EF0900000
000000005929009000E0EA4E1F4E5E90E1E2
F3F0E5E9E3000E5F2E0590900,623
1090 DATA 00000000005941030505050505
2000505050505050505050505004C09000
0000000050505050505050505,470
1100 DATA 0505050505050505050505050505
0505050500000000005152525252525252525
5252525752525252525252525252,112
1110 DATA 52525252524500000007C0C0
A1EEFAE1C0EF041F342EC0E0E007CA1E4F2E
F3E0900400000007C000000,309
1120 DATA 004152525252525252525252525252
525252525252525252525252524000000
7C1C00919298042F9F4E300,430
1130 DATA 1E7C007A757200216472057373
007C0000000007C0600009A0A2F9F4E5000010
7C00650696E737465A6C6C7560,717
1140 DATA 67007C0000000007C0A000091600A2
F9F4E500000047C00721074672736503606656E
007C0000000054525252525252,402
1150 DATA 525252525252525252525252525252
52525252525252524300000009E4F5F2E3E0007
009360710FAF5F200A0E1F0,124
1160 DATA F0F4E1F5F3F7E1E0EC5900490555
5555555555555555554F09000000600000000
00000059400555055050505,717
1170 DATA 0905050534CF442EF43A104F2E9F3
F3E5009FACEFE1E4000AEE9ECE5E3E1F6E30040
E9ECE5905F2F2DFF200000000,492
1180 DATA 0005E3E0F4EFF29F0009A0E9E7E9
EEEE0000000000590945EEE4E500000000000
0059000051525252525252,249
1190 DATA 52525252525257525252525252525252
525252525245000000007C1C0E0EA1EEFA01E0
EC0E0E007C0005E3E0F4EFF2,996
1200 DATA 9A0400060007C0000000415252
5252525252525252525252525252525252525252
5252525252440000000007C1C,330
1210 DATA 00029000005E3E0F4900A1E7C003A
7572005A56074AF720007C00000000007C0000
00940003E5E0F4000137C00,160
1220 DATA A5696E73746A6C6C756E67007C00
000007C000000091000E3E0F490000070007
330764F0430044727565636E0,732
1230 DATA 656E007C000000005A5252525252
525252525252525252525252525252525252
525243000000009E4F5F2E3E0,753
1240 DATA 007A505A30700FAF5F200DE1F5
F0F4E1F5F3F7E1E0EC0900000051525252525252
52525252525252525252525252,990
1250 DATA 525252525245000000007C0A4E9
F2E9E3E4CFF2F900CFEA00AF2E9F6E500A1C3
7C00005152525252525252525252,49
1260 DATA 52525252525252525252525252525252
52905A52457C000000000000000000000042
5600000000000000000F0000,876
1270 DATA 007C5A525252525252525252525252
525252525252525252525252525245050409
F3E0DCA5F2F2EFF200AEF20E,520
1280 DATA 0000A059009000AEEF00A009ECE5F3
0001D1004B5A9044EC44E7453130A2A2E2A9044
313A46494C434EA140452E45,537
1290 DATA 5A5490424552424554A0042100100
00001279710000000000E9E0E0A479644C247
C5479046A0473C475A473F47,724
1300 DATA 53474247564790463200074A0F4A
004AC24A0C4AC540FF040A04A04000000000
0600100640100640010040,263
1310 DATA 0100640010006401900A0010060
0100001500130013001300A7CA10IFEE0940
3003EE04400400CBCB32000C,448
1320 DATA A9760040000940C0A400A00A00
0203060A70E040A6900854000602A9000E02
4C5FE4A00CA000DE745900C0,435
1330 DATA 45A9CA424590C0A00A2910A09CA
0E4402F0032000150A0FF9A0510297F0510D0
0E02A90000F9000A4A1A21F,263
1340 DATA 00004900091A0C0002045E346F900
FF9000SDE6A1CA10EF001F20A9000005A054004
A1A90000E0200440A21700002,38
1350 DATA 4590E030CA10F7A930000F002A200
0600A20306A7A9039200300046659000003042
4590C502A9009C00C3CA10E7,449
1360 DATA A210A000009FF2030440A25049C090
2534490390252702A10F309300070A2028E10
00C00E6F02C00A41A9399D2F,519
1370 DATA 02A9760040000940C00A0A900600
72A045205C0E4205A4CA90CA24000300020E3100
A900005AF0500005A009ED0CA0,194
1380 DATA A9C0A000E0EA400CA0902020A0A0
81A00A9FF203440A200A0062034400A034A9E0
990734A907990797060DF3A0,495
1390 DATA 200A9E7999154A9FF999133999136
A9E79991370E00E0A9610C30D220364920FF4C
C95130160935E446DF471012,992
1400 DATA 5AE9310A0391A450A00910450S
416C00002040A0AC0046100120A2E30C3E20
45404F00A5AC2A544C2A5640,003
1410 DATA 20934020544C20FF4C02A07E3FE46
F00C410F020404C1947D0A0A02224505A0
5023430A16C00A0A9012009,610
1420 DATA 002CA900106A5A005AA90100A004C
6547A9062CA900C2A9000A4500000E004050
9003CA00A9000D1F00C1347,702
1430 DATA A5AC0A05A20020A2034020344003FF
4CC91AF03FC99F030C930900CC947000C941
000AC93A90A0A20040404C7947,003
1440 DATA 38E930C9009002E907A0040A6A02A
A00E0DF9A0AA05002044020544C2093484C79
474C06474C9046A9FF20A900,412
1450 DATA E0A92045402A911A5A9300A21A20
5A40A00CAF00995A1F0B10E04700A0FA000004
A4A9F00630IF2036A9A6A4A9,434
1460 DATA F0A9130IF20FF4CC910F0E406A490
F544F00CC99F040C97EF030C926F020E99030
062000A0E4DC947C950900CC9,671
1470 DATA 50A000C94100A0C9300A0A204040
4CE047E00CF0F4E000100A0C9419DEE20C04009
00903016EA0A1007C600000,939
1480 DATA 90501F004A04DE047206AA600A
05A93002A0020050E10092000F4E209E4C4CC9
470509300004240306E2000E,879
1490 DATA 4C5509AC904A4210040907904405A9
50A0430032000E4030A9FFC0D05A0000C70015A
F07A09029344B3A9509304503,409
1500 DATA 5AA2E302E9A0A0A0A0A0E9E9E50A0A0
201248100A0C9A0A0A0A330900A9A0A7300A95A0A
A930050372490505AE20924C,374
1510 DATA 2A4540000CA9009701F0B90C0007
4921A399900F00200439900IF0810EC0090205A
4C400A5AE20204CA54020500,201
1520 DATA 4EA000A0027100F544300EC92EF007
20C0400A001FE0C010E04C00470090A0300A0
844EA21004900904405093C90,100
1530 DATA 45020A0042010401007C00060A74C
A54A0A0290A0030ME9145033A000230E000SC
900A0033CE0013C00A3906S,274
1540 DATA A0054090653AE05AEC5A0300960A0
0210CFA00AAE0000IC000A21020124010A0C000
C010001510009A900A0300500,673
1550 DATA 0A0A05A0300AC5A0EC0A0AC0CE491D
532020420300A3E0400A0045AF00070A000034C
324420754C2036A049000050,347
1560 DATA 0004050C05030500E4000005AF0A00
0004050E00035A901500100A3A9520000305A900A0
A0A900000003A9000E04903,002
```

1 Ohrenschmaus: Soundmachina

[illegible]

ATARI 400 130 XE [illegible]

Best.-Nr. AT 1 29.80 DM

2 Lehrreich: Das Assemblerbuch

[illegible] ATMAS II Assembler [illegible] DIN A5 [illegible]

Best.-Nr. AT 10 29.80 DM

3 Vielfalt: Atari Power Superbuch

[illegible] Tips & Tricks [illegible] DIN A4 [illegible]

Best.-Nr. AT 3 29.– DM

4 Nachschlag: Die Hexenküche

[illegible]

Best.-Nr. AT 4 29.80 DM

5 Zuschlag: Disk zu Hexenküche

[illegible]

Best.-Nr. AT 5 18.80 DM

6 Fix: ATMAS-II Macro Assembler

[illegible]

ATARI 400 130 XE

Best.-Nr. AT 6 Diskette 49.– DM

7 Geistsparend: ATMAS Toolbox

[illegible] Auf Diskette mit Anleitung [illegible]

ATARI 400 130 XE 48/64 K

Best.-Nr. AT 7 19.80 DM

8 Praktisch: Monitor XL

[illegible] Directory-Anzeige [illegible] DOS [illegible]

ATARI 600 XL (64 K)/800 XL/130 XE

Best.-Nr. AT 8 16.80 DM

9 Brandneu: Design Master

[illegible] deutsche Anleitung.

ATARI 600 XL (64 K)/800 XL/130 XE

Best.-Nr. AT 9 Diskette 18.80 DM

take 9 !!

CK Spezialitäten-Bestellschein

Anzahl	Best.-Nr.	Artikel	
	AT 1	Soundmachine	
	AT 10	Das Assemblerbuch	
	AT 3	Atari Power Superbuch	
	AT 4	Die Hexenküche	
	AT 5	Disk zur Hexenküche	
	AT 6	Macro Assembler	
	AT 7	ATMAS Toolbox	
	AT 8	Monitor XL	
	AT 9	Design Master	

Ich wünsche folgende Bestellung:

☐ Nachnahme

☐ Vorauskasse

[illegible]

Name des Bestellers ____ PLZ/Ort ____

Straße ____ Datum/Unterschrift ____

Coupon bitte [illegible] auf Postkarte kleben und einsenden an:

Computer Kontakt, Softwareversand, Postfach 1640, 7518 Bretten.

```
1570 DATA [illegible],594
1580 DATA [illegible],227
1590 DATA [illegible],524
1600 DATA [illegible],291
1610 DATA [illegible],445
1620 DATA [illegible],314
1630 DATA [illegible],47
1640 DATA [illegible],83
1650 DATA [illegible],96
1660 DATA [illegible],582
1670 DATA [illegible],454
1680 DATA [illegible],285
1690 DATA [illegible],593
1700 DATA [illegible],585
1710 DATA [illegible],244
1720 DATA [illegible],175
1730 DATA [illegible],562
1740 DATA [illegible],474
1750 DATA [illegible]
1760 DATA [illegible],621
1770 DATA [illegible],793
1780 DATA [illegible],423
1790 DATA [illegible]
1800 DATA [illegible],684
1810 DATA [illegible],880
1820 DATA [illegible],252
1830 DATA [illegible],935
1840 DATA [illegible]
1850 DATA [illegible],533
1860 DATA [illegible],589
1870 DATA [illegible],250
1880 DATA [illegible],532
1890 DATA [illegible],934
1900 DATA [illegible],215
1910 DATA [illegible],730
```

Beispiel

```
100 POKE 106,PEEK(106)-4:GRAPHICS 0:RE
STORE 140
110 DIM T$(29):FOR I=1 TO 29:READ D:T$
(I)=CHR$(D):NEXT I
120 CLOSE #1:OPEN #1,4,0,"D:ZEICHSAT A
TZ":T=USR(ADR(T$)):CLOSE #1
130 POKE 756,PEEK(106):CLR
140 DATA 104,162,16,169,7,157,66,3,165
,106
150 DATA 157,69,3,169,0,157,68,3,157,7
2,3
160 DATA 169,4,157,73,3,76,86,228
```

BASIC-Zauberer

Die Super Hilfsdiskette für Ihren ATARI!

Über 40 Files helfen Ihnen bei vielen Programmierproblemen - jeden Tag!

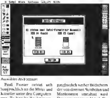
BASIC-Zauberer - das sind:

* BASIC-Erweiterungen wie
-Autonumber
-REM Killer
-REM Marker & -Entmarker
-COLOR-Maker
-HEX BIN DEZ -Wandlung
-Block Zeilen-Delete
-Variablen-Lister
* Tolle Hilfsprogramme wie
-Zeichensatzeditor
-Farb-Zeicheneditor
-TT-Titelmaschine
* nützliche MS-Programme wie
-Zeichensatz LOAD/SAVE/COPY
-128 Farben-Effekt
-Bilder-Laderoutine
und vieles mehr! nur DM 29.-

Thomas Tausend, Ried 17
Tel.: (09363)/1664
8961 Sulzberg

Profi Painter – für den Atari ST

Auch das jüngste Kind der Data Becker Erfolgsreihe wurde auf den Atari ST umgesetzt. Einschränkend muß aber gleich gesagt werden, daß der Autor die beiden Vorbilder (CPC und C 64) nicht kannte und auch einen anderen Namen vorgegeben hatte. Weder Programmidee noch die grundsätzlichen Funktionen sind also abgekupfert. Vielmehr entstand erwartungsgemäß ein Programm, das von den Möglichkeiten und Fähigkeiten her jeden Konkurrenten wie einen Ritter mit links aus dem Felde schlägt.


Datei Block Optionen Schrift Hilfe
Datei eröffnen
Es stehen zwei Datei-Formate zur Auswahl:
DIN A4 (hoch)
DIN A5 (quer)
DIN A4
DIN A5
ABBRUCH
Profi Painter ST

Auswählen des Formats

Profi Painter richtet sich hauptsächlich an die Maler und Künstler unter den Computeristen. Technische Zeichnungen, Formulare usw. sollten von "Easy Draw" oder "Gem Draw" erstellt werden. Für Bilder aller Art, Plakate oder auch Hintergründe von Spielen ist der Einsatz des vorliegenden Programms jedoch ein Muß.

Nach dem Laden von Profi Painter kann der Anwender zwischen zwei Formaten wählen, in denen sein Bild entstehen soll. Man wundert sich allerdings, daß außer DIN A4 Hoch und DIN A5 Querformat keine weiteren Formate möglich sind. Von einem Programm, das auf so einem schnellen und reichlich mit Speicherplatz ausgestatteten Computer läuft, kann man wenigstens noch DIN A4 quer und eigentlich auch DIN A3 quer verlangen. Man sollte meinen, daß 1 MByte dazu ausreicht. Aber es ist ja noch nicht aller Tage Abend. Denn wie der Hersteller verlauten ließ, sind in Zukunft noch mehrere überarbeitete Versionen zu erwarten. Da aber noch weitere Schwachpunkte im Programm zu finden sind, die wir im folgenden noch näher betrachten werden, hätte man lieber noch ein paar Monate warten sollen, um dann sofort ein ausgereiftes Produkt auf den Markt zu bringen.

Nach der beschriebenen Voreinstellung offenbart sich ein jungfräulich weißer Bildschirm, der von diversen Symbolen und Menünamen umrahmt wird. Das erste spontane Urteil: Optisch sehr ansprechend und sorgfältig aufgebaut. Auch beim praktischen Arbeiten entpuppt sich die Bildschirmaufteilung als sehr sinnvoll. Die rechte Leiste zeigt Symbole mit den Mal- und Zeichenfunktionen, die verschiedenen Strichstärken von GEM, den Malstatus und das aktuelle Muster an, das in der linken Leiste geändert werden kann. Hier sind alle zur Verfügung stehenden Füllmuster dargestellt, wobei den erfahreneren Freaks sofort auffällt, daß nur die Standardmuster von GEM vorhanden sind. Große Enttäuschung macht sich gar breit, wenn deutlich wird, daß das Erstellen eigener Füllmuster nicht möglich ist. Dies ist sicherlich ein sehr gravierender Fehler, den der Konkurrent „Degas" nicht aufweist.

In der obersten Zeile finden wir schließlich wie üblich die Menüs von Profi Painter, mit denen wir uns gleich noch genauer befassen werden. In der Mitte thront das Zeichenfenster, das beliebig vergrößert und verkleinert werden kann, wie wir es schon aus der Diskettenverwaltung gewohnt sind.

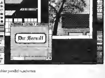
Die freie Wahl des Bildformats hat auch einen ganz konkreten Hintergrund, denn mit dem Programm kann man sage und schreibe drei Bilder gleichzeitig bearbeiten. Mit den richtigen Einstellungen sind die Kunstwerke auch gleichzeitig sichtbar, und mit den Blockoperationen können Objekte zwischen den Files ausgetauscht werden. Dies erweist sich im praktischen Einsatz schnell als sehr nützlich, denn oft sind in einem Bild Objekte so gut gelungen, daß man sie auch in Zukunft weiter verwenden möchte.

Diese Blockmanipulationen beginnen zunächst mit der Definition des gewünschten Bildschirmbereichs, was mit dem bekannten Auswahlrahmen realisiert wird. Nun kann man den Block, der immer eine quadratische Form hat, löschen oder auch versetzen, indem man ihn einfach mit dem Mauszeiger an die gewünschte Stelle schiebt. Die ruckhaften und flimmernden Bewegungen des Bildschirmbereichs stören dabei allerdings sehr. Selbst kleine Verschiebungen bewirken ein Flimmern, das ein genaues Arbeiten fast unmöglich macht und dem Anwender eigentlich auch nicht zugemutet werden kann. Der Grund für diese und andere Geschwindigkeitseinbußen ist sicherlich in der Tatsache zu suchen, daß Profi Painter in "C", also einer Hochsprache programmiert wurde, und dort wirken sich eben die Geschwindigkeitsunterschiede zum reinen Maschinencode verheerend aus.

Doch zurück zu den Blockoperationen. Der definierte Bereich kann natürlich auch kopiert und an beliebiger Stelle neu eingesetzt werden. Diese Stelle kann, wie bereits erwähnt, auch in einem anderen Fenster liegen, das gleichzeitig eröffnet wurde. Wem diese Form der Objektübertragung zwischen zwei oder mehreren Files zu umständlich ist, der kann den Block auch unter beliebigem Namen auf Diskette abspeichern und jederzeit in einem anderen Bild wieder einladen.

Neben diesen ganzen Möglichkeiten des Versetzens, Löschens und Kopierens kann auch der Block an sich verändert werden. Invertieren, horizontales und vertikales Spiegeln sowie das Drehen um 90 Grad sind auf Mausknopfdruck realisierbar.

Die Zeichenfunktionen werden in der linken Leiste durch Symbole dargestellt. Selbstre-



3 Bilder parallel bearbeiten

dend kann Profi Painter Kreise und Rechtecke mit oder ohne gerundete Ecken in beliebiger Form an beliebiger Stelle und auf Wunsch mit den aktuellen Mustern gefüllt darstellen. Das Programm kann Linien ziehen und erlaubt auch das Freihandzeichnen. Dabei unterscheidet man aber noch zwischen Pinsel und Bleistift. Letzterer soll für feine Arbeiten eingesetzt werden, während der Pinsel in 24 verschiedenen Formen zur Verfügung steht und die gröberen Vorarbeiten übernimmt.

Weniger selbstverständlich, aber nur Fast-Standard ist die Sprühdose für die friedlichen Graffitis am Computer. Es stehen drei davon in verschiedenen Größen zur Auswahl. Deren Funktion läßt jedoch zu wünschen übrig. Bisher verstand man unter sprayen, daß sich unter dem Mauszeiger Punkte bilden und sich die einzufärbende Fläche wie beim Einsatz einer richtigen Spraydose durch kontinuierlichen Tastendruck immer mehr verdichtet. Profi Painter fällt hier aus der Reihe. Der Tastendruck bewirkt nur den einmaligen Abdruck des Spraymusters. Weitergehende Effekte müssen durch Hin- und Herbewegen erreicht werden, was allerdings ein wenig verschmiert und unsauber aussieht. Daher ist der Sinn der Spraydose ein wenig fraglich, läßt sich derselbe Effekt doch mit dem Pinsel der entsprechenden Form erzielen.

Daneben gibt es noch das Füllen von Flächen und die etwas zweifelhafte Radiergummifunktion, mit der sich nur sehr große Teile der Zeichnung entfernen lassen. Feinarbeit wird mit der Lupe gemacht, und hier bietet Profi Painter eine nachahmenswerte Neuerung. Nach dem Anklicken des Lupensymbols erscheint in der linken oberen Ecke der Rahmen für den vergrößerten Bildteil und in der linken unteren Ecke das Fenster mit der Vergrößerung. Der Rahmen kann nun beliebig verschoben und gesetzt werden, im Lupenfenster scrollt die Pixeldarstellung mit. Sitzt das Lupenfenster nicht ganz richtig, kann es wie ein ganz normales Window versetzt werden. Ist der richtige Ausschnitt erreicht, können die Pixel einzeln gesetzt und gelöscht werden, wobei die Änderung auch gleich im Gesamtbild sichtbar wird.

Eine weitere Besonderheit, die vollkommen aus dem Rahmen fällt, ist die Textoption. Die Möglichkeiten, die sich hier eröffnen, sind faszinierend. Neben acht Größen stehen die Schrift-Typen fett, outlined, unterstrichen, kursiv und hell zur Verfügung, die man natürlich auch noch untereinander mischen kann. Außerdem ist das Laden von sechs weiteren GEM-Zeichensätzen möglich, zwischen denen man ganz einfach hin- und herschalten kann. Schier unglaublich ist, daß der Text in vier verschiedenen Richtungen geschrieben werden kann. Beschriftungen auf dem Kopf oder quer sind für das Programm kein Problem.

Erwähnenswert wären noch die vier Zeichenmodi, in denen man arbeiten kann. Gezeichnetes wird von DECKEND gnadenlos überdeckt, TRANSPARENT läßt darunter liegendes durchschimmern, XOR führt eine XOR-Verknüpfung der beiden Flächen durch, und bei INVERS zeigt sich Weißes schwarz und Schwarzes weiß.

Nachdem ein Bild mit dieser Fülle von Funktionen erstellt wurde, soll es natürlich auch abgespeichert und nach Möglichkeit ausgedruckt werden. Beim Speichern stehen zwei Formate zur Verfügung, die nach Belieben eingestellt werden können. Auf diese Weise sind Profi-Painter-Bilder ohne Probleme kompatibel zu "Degas" und "Doodle", so daß man beispielsweise ein Bild mit "Degas" weiterbearbeiten kann, indem man ein selbstdefiniertes Muster einfügt.

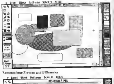
Der Ausdruck erfolgt entsprechend der Einstellung des Blattformats maßstabsgetreu. Dabei bedient sich Profi Painter einer Druckerdatei, die man jederzeit mit einem Texteditor ändern kann. Hier werden die Grafikcodes einzeln angegeben, so daß auch der No-Name-Drucker aus Hong Kong mit den entsprechenden Einstellungen Grafik ausspuckt.

Profi Painter besitzt also einige beeindruckende Funktionen wie die Lupe und die Textoption, die in dieser Form einmalig und vorbildlich sind. Dagegen ist die Geschwindigkeit an einigen Stellen nicht besonders hoch, auch eine freie Musterdefinition ist nicht möglich.

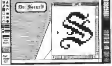
Man vermißt mehrere Blattformate, und auch das Handbuch verrät nur das, was der Anwender unbedingt wissen muß. Wer mit diesen Mankos leben kann, darf sich an einem optisch ansprechenden und in der besprochenen Weise sicherlich leistungsfähigen Programm erfreuen. Den anderen Anwendern sei empfohlen, auf neue Versionen zu warten, denn neben vielen Verbesserungen soll auch noch ein Profi Painter in Farbe erscheinen. Trotzdem: Profi Painter kostet 99,– DM und bietet für einen knappen Hunderter relativ viel. Die leistungsfähigere Konkurrenz (z.B. "ST Art") ist schon im Anmarsch, wird aber sicherlich um einiges teurer sein.

System	Atari ST mit SM 124
Hersteller	Data Becker
Preis	99 DM
Bezugsquelle	Data Becker

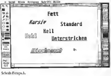
Thomas Tai



Verschiedene Formen und Füllmuster



Lupe und Auswahlrahmen (gestrichelt)



Schrift-Beispiel

Dem Geld auf der Spur

Ein Programm für den Atari ST

Besonders Computerfreaks kennen dieses erschreckende "Wo ist nur mein Geld geblieben"-Gefühl. Am Anfang des Monats sah es noch recht gut aus, und nach zwei Wochen bahnt sich eine chronische Ebbe in der Kasse an. Ein Weg, dieser Entwicklung entgegenzuwirken, ist sicherlich, das schwer verdiente Geld an eine Vertrauensperson (Ehefrau, Kinder etc.) weiterzugeben, die sicherlich nie im Leben einen Pfennig zu viel ausgeben würden. Wer allerdings kein "Vertrauen" zu irgend jemandem hat (bei Geld hört ja bekanntlich die Freundschaft auf), wird den Computer zu Rate ziehen. "ST Heimfinanz" ist nun ein komfortables Programm, mit dem die laufenden Ausgaben überwacht, in einer monatlichen Übersicht festgehalten und über einen längeren Zeitraum statistisch ausgewertet werden können.

Der Beginn der Arbeit mit dem Programm ist von diversen Voreinstellungen gekennzeichnet, die auf der Diskette abgespeichert und nie wieder erneuert werden müssen. Neben dem Benutzernamen, dem aktuellen Kassenstand sowie dem Monat und Jahr, in dem die Überwachung beginnen soll, muß man ein Paßwort eingeben, das die Daten vor fremdem Zugriff schützt. Vergessen darf man den Code allerdings auch nicht, denn dann kommt man selbst nicht mehr an sie heran. Aber das ist nun mal der Tribut, den man der Datensicherheit zollen muß. Es ist erstaunlich, wieviel Mühe der Programmierer in den Schutz der Angaben gelegt hat. Die Daten sind auf Diskette so abgelegt, daß ein Manipulieren über Diskmonitor oder sonstige Hilfsmittel nahezu unmöglich ist.

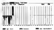


Bild 1: Monatsbilanz

Nach den Voreinstellungen müssen die einzelnen Ausgabenposten benannt werden, die regelmäßig oder häufig anfallen. Um die Übersicht zu behalten, sind nur 27 Posten möglich, so daß man gezwungen ist, die Ausgaben in größeren Gruppen

(z.B. Nahrungsmittel) zusammenzufassen. Haushalte, die ihr Geld häufig in sehr verschiedene Dinge stecken, werden dies vielleicht zunächst bemängeln, doch bei der Erstellung von Statistiken wird man schnell vom Gegenteil überzeugt, da hier eine sinnvolle Auswertung nur mit einer begrenzten Anzahl von Ausgabenposten möglich ist.

Nun kann man die harten Zahlen den jeweiligen Posten zuordnen. Leider liegt bei dieser Eingabe einer der gravierenden Schwachpunkte von "ST Heimfinanz". Zunächst muß die jeweilige Ausgabe angeklickt werden, dann erscheint noch einmal ein Fenster, in dem die Eingabe von Geldbeträgen bestätigt wird, und erst dann kann man wirklich loslegen. Besser wäre es gewesen, nur das entsprechende Feld anzuklicken und die Eintragung gleich auf dem Kontenblatt zu vollziehen. Bei einem Arbeitsschritt wie diesem, der sehr häufig genutzt wird, ist das Durcharbeiten durch Untermenüs, und sei es nur ein einziges, sehr lästig. Auch die Zahleneingabe ist nicht sonderlich komfortabel gestaltet. So müssen Zahlen ohne Nachkommastellen trotzdem ein ".00" enthalten. Ansonsten wird die Eingabe nicht akzeptiert. Auch dies ist ein Fall, der häufig eintreten wird und lästige Tipparbeit beschert, die nicht sein müßte. Vorteilhaft bei der Eingabe ist aber der sog. "Limitbetrag", den man vor der Arbeit mit dem Programm eingibt. Wird dieser Wert mit einer Ausgabe überstiegen, warnt "ST Heimfinanz" mit einem Alert-Window. Wie nützlich diese Warnung ist, ist eine andere Sache.

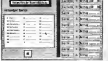
Die Arbeit mit "ST Heimfinanz" gestaltet sich nun so, daß der Benutzer von Zeit zu Zeit die ausgegebenen Beträge eintippt, zum Monatsende einen Abschluß macht und einen neuen Monat anfängt, der wieder separat abgespeichert wird. Dabei ist es sehr nützlich, daß regelmäßig anfallende Ausgaben wie Miete, Raten oder Beiträge in eine automatische Monatsbuchung aufgenommen werden können. Zu Beginn jeden Monats klickt man "Fixkosten buchen" an, und die Beträge, die sowieso jeden Monat anfallen, werden automatisch abgebucht.

Nun reicht es natürlich nicht, die Ausgaben und Einnahmen fein säuberlich in den Computer einzugeben. Viel wichtiger ist die Auswertung, mit der man ja auch erst dem sprichwörtlichen Loch im Geldbeutel auf die Spur kommt. Die Auswertung mit "ST Heimfinanz" kann in tabellarischer und grafischer Form auf Bildschirm und Drucker ausgegeben werden. Dabei wird der Stand eines gewählten Monats mit dem Vormonat verglichen, ein Jahresdurchschnitt ermittelt, die Kassendifferenz zwischen Beginn und aktuellem Zeitpunkt und die Ausgaben und Einnahmen insgesamt errechnet. In einer Tabelle hat man zwar die genauen Werte für die Ausgaben, doch übersichtlicher und optisch ansprechender ist eine Grafik. Das Programm gibt sie auf dem Bildschirm aus, aber auch eine Hardcopy ist möglich. Die Ausgabenposten in Tabelle und Grafik können nach Namen, Beträgen und Nummern sortiert werden.

"ST Heimfinanz" ist als komfortables und leicht zu bedienendes Programm für den Hausgebrauch einzustufen. Umfangreiche Kalkulationen lassen sich mit dem Programm natürlich nicht erstellen, aber das verlangt ja auch keiner. Beachtet werden müssen nur einige Bedienungsklippen, die einem mit der Zeit sehr schnell auf den Wecker fallen werden.

System Atari ST
Bezugsquelle
Ingenieur Büro
Harald Zoschke

Thomas Tu



Fixkosten buchen

Das große LOGO-Buch zum Atari ST

Von Dr. Gerhard Sauer
Data Becker Verlag
297 Seiten, 49,– DM
ISBN 3-89011-122-X

LOGO ist in Amerika schon sehr lange als leicht zu erlernende Programmiersprache auch für Kinder bekannt. In Deutschland fehlte bisher der richtige Durchbruch. Dennoch lieferte Atari zum ST zunächst LOGO kostenlos mit aus. Leider wurde diese Beilage inzwischen mangels Nachfrage eingestellt und Interessierte müssen LOGO jetzt käuflich erwerben.

Ein Faktor, der LOGO zum Durchbruch verhelfen könnte, ist sicherlich das vorliegende Buch. Es ist ein hervorragend geschriebenes Lehrbuch, das die Seiten nicht mit aus dem Handbuch abgeschriebenen Befehlsübersichten füllt, sondern auf 400 Seiten Erklärung pur bietet. Es richtet sich sowohl an den totalen Computerneuling als auch an erfahrene Freaks, die in LOGO einsteigen wollen.

Das Buch ist in 18 Lektionen aufgeteilt, von denen jede neue Sprachelemente erklärt und an Beispielprogrammen verdeutlicht. Der Autor beginnt mit dem Laden und Starten von LOGO und geht auf die Handhabung der Windows ein. Danach beginnt der Einstieg in die Sprache. Beginnend mit den einfachen Sprachelementen breitet er Stück für Stück alle Geheimnisse von LOGO vor dem Leser aus. Er entwickelt daher Programme, die den speziellen Eigenschaften von LOGO angepaßt sind und mit der steigenden Komplexität der erklärten Befehle ebenfalls leistungsfähiger und schwerer werden. Der Autor legt daher sehr viel Gewicht auf die ausführliche Behandlung der Grafikfunktionen und der Realisierung von Künstlicher Intelligenz. So entwickelt er in einem Kapitel das bekannte selbstlernende Programm, das Tiere anhand von bestimmten Kriterien errät.

Zu Beginn jeder Lektion wird übersichtlich dargestellt, welche Sprachelemente im folgenden neu eingeführt werden, und am Ende findet der Leser einige Übungsaufgaben, anhand derer er sein Wissen überprüfen kann. Da aber leider keine Lösungen zu den Aufgaben vorhanden sind, muß man hier den einzigen Schwachpunkt des Buches aufdecken. Obwohl sich die Aufgaben teilweise sehr stark an den behandelten Stoff anlehnen, sind vor allem Laien oft unsicher und wären für die Lösungen sicherlich dankbar gewesen.

Das LOGO-Buch zum Atari ST ist das Standardwerk für alle, die den Computer der neuen Generation in dieser Sprache programmieren wollen. Es ist gut geschrieben und immer leicht verständlich. Wenn auch die Aufmachung und einige Stellen des Textes sehr an Schulbücher erinnern, sollte man sich doch nicht abschrecken lassen, denn es handelt sich nun mal um ein Fachbuch. Für eine eventuelle 2. Auflage sei aber empfohlen, den Text lockerer zu gestalten.

Thomas Ta

Computer-Kontakt
das Heft mit den
preisgünstigen
Kleinanzeigen

Das große Basic-Buch zum Atari ST

Von Frank Kampow und Norbert Szczepanowski
Data Becker Verlag
268 Seiten, 39,– DM
ISBN 3-89011-121-1

Basic auf dem Atari ST ist ein heikles Thema, da der mitgelieferte Interpreter einige Macken und Fehler hat, die vorher hätten beseitigt werden sollen. Trotzdem wurde der Anwender mit einer fehlerhaften Version abgespeist, und wenn er nicht gewillt ist, für die anderen käuflichen Interpreter Geld auszugeben, muß er einfach aus dem Vorhandenen das Beste machen. Dabei wird auch das vorliegende Buch eine große Hilfe sein.

Allen Unkenrufen zum Trotz kann man sagen, daß sich ein Anfänger die Logik des Programmierens am einfachsten und schnellsten mit Basic beibringt. Das war aufgrund der spärlichen Dokumentation des ST Basic nur schlecht möglich, und genau hier setzt das Basic-Buch zum ST an. Es wendet sich zunächst an den völligen Neuling, der neben Anwenderprogrammen auch eigene Problemlösungen auf seinem Computer laufen lassen will, vom Programmieren aber kein Ahnung hat. Behutsam wird er in die Grundlagen eingeführt. Begriffe wie Dualsystem, Hexadezimalsystem, ASCII-Code oder Bit und Byte bekommen einen konkreten Hintergrund. Außerdem wird das Flußdiagramm sowie seine Nutzung und Bedeutung eingeführt. Dann geht es schon los mit der Programmierung.

Der Leser erstellt sein erstes Programm und im folgenden noch viele andere kleine Listings, die die Funktion der grundlegendsten Befehle verdeutlichen. Nachdem die Autoren diese Grundlagen gelegt haben, kommen kompliziertere Strukturen an die Reihe, wobei die kleinen Beispielprogramme teilweise mit Flußdiagrammen erarbeitet werden. Zu jedem dieser Kapitel werden dem Leser Aufgaben gestellt, an denen er seinen Lernerfolg überprüfen und eventuelle Unklarheiten erkennen und nacharbeiten kann.

Die folgenden Kapitel behandeln komplexere Programmiertechniken. Von mehrdimensionalen Feldern und der Unterprogrammtechnik über einen sehr kurzen Sprung in die Grafik- und Soundprogrammierung bis hin zum Aufruf von GEM-Routinen wird der Bogen gespannt, wobei verständlicherweise nicht alles in großer Ausführlichkeit zur Sprache kommt. So sollte man die Kapitel über Grafik, Sound und GEM nur als kleinen Appetitanreger betrachten. Ernsthafte Anwendungen sind mit dem vermittelten Wissen kaum möglich.

Der Anhang hat es dann noch einmal in sich. Eine Übersicht listet alle Befehle in alphabetischer Reihenfolge mit Format, Erklärung der Funktion und einem kleinen Beispiel auf. Außerdem werden die Fehlermeldungen in Deutsch erklärt.

Das Buch ist eine hervorragende Einführung in Basic selbst für den totalen Neueinsteiger. Ein guter didaktischer Aufbau mit wenig Fachchinesisch und weitere Merkmale des Werkes. Mit dem nützlichen Nachschlageteil bleibt es auch nach dem Durcharbeiten noch interessant und wird sich beim täglichen Programmiereinsatz bewähren.

Thomas Ta

Tom Rowley

Sprühende Ideen mit Atari Grafik

SPRÜHENDE IDEEN

150 Seiten
Sprühende Ideen ist ein Lehrbuch, das mit den Grafikmöglichkeiten des ATARI in der Leistungsklasse von Disketten in Farbgebung und der Entwicklung von Bildschirmeinheiten [illegible] führt. Es [illegible] Leser genug [illegible] Kenntnisse der Programmiersprache Basic, auch wenn das Buch gelegentlich [illegible] für die [illegible] zeigt.

Bestellnummer TW 104 **DM 44,–**

C. Lorenz

Das große Spielebuch für Atari

Das Grosse Spiele-Buch
ATARI

200 Seiten
Dieses Buch enthält eine Reihe aktueller Programme für die Atari 600 XL und 800 XL und ist eine Weiterführung von Buch 1. Dem großen Spiele-Buch für Atari. Es bringt eine Reihe neuer Spiele, Programme zur Soundeinstellung und ein Kapitel über Grafik, Spielereien mit dem Atari. Außerdem enthält es einige Tips und Programme zum Zeichensatz des Atari.

Bestellnummer H 926 **DM 39,00**

C. Lorenz

Das große Spielebuch für Atari

Das GROSSE
ATARI

[illegible] Seiten
Aufregende Computerspiele in Atari-Basic. Neben Spielen finden Sie hier eine Reihe hochinteressanter Anregungen für eigene Programme, 3D-Grafik, Bewegung und Struktur, Grafik und Ton in FORTH, Tonprogrammierung usw.

Bestellnummer H 8[illegible] **DM 39,00**

Handbuch für Hacker

In diesem Buch finden Sie endlich alle Antworten auf die Fragen, die sich jeder stellt und bislang keiner beantworten konnte. Mit vielen Nummern und Datenbanken. Endlich wird hier erklärt, wie man mit Daten [illegible] arbeitet. Viele Tips und Tricks zum Mailboxbetrieb. Steigen Sie mit ein in die Unterwelt der DFÜ.

Bestellnummer 3 **DM 58,–**

Raeto West

C 64/SX 64 – Computer Handbuch

600 Seiten
Insgesamt ein Jahr verwendete der Autor Raeto West mit der Analyse und Dokumentation auf den C-64. Das Ergebnis war ein Nachschlagewerk, das die [illegible] Buch [illegible] Computer [illegible] gibt [illegible] Fakten und [illegible]. Es enthält über 100 Programmierungen aller Mal-Funktionen, sowie die schwierigen, seltenen und meist gemiedenen

Bestellnummer TW 102 **DM 69,–**

Don Inman / Kurt Inman

Der Atari Assembler

Der ATARI Assembler

276 Seiten, 82 Abb.
ca. 100 Programme
Mit diesem Buch können Sie das Programmieren in Assembler lernen und sich gleichzeitig mit der Anwendung [illegible] Atari Assembler Moduls auf Ihrem Atari 400- oder 800-Modell vertraut machen. Das Buch ist eine ausgezeichnete Einführung für Leser mit einigen Grundkenntnissen in Basic, setzt aber keinerlei Assembler-Kenntnisse voraus.

Bestellnummer [illegible] 202 **DM 39,–**

Karl Heinz Koch

ATARI Spiele programmieren

ATARI
Spiele programmieren

[illegible] Seiten
Das Buch führt Schritt für Schritt in das Programmieren in BASIC ein. Dabei werden anhand von [illegible] Befehlen [illegible] erklärt. [illegible] Befehle und ihre Wirkung [illegible] gemacht. Auf Verständlichkeit wird besonderer Wert gelegt, was für Käufer der [illegible] keine Schwierigkeiten [illegible].

Bestellnummer [illegible] 507 **DM 38,–**

Julian Reschke / Andreas Wiethoff

Das ATARI Profibuch

29[illegible] Seiten. Das richtige Buch für Sie, wenn Sie einen ATARI 400, 800, 800 XL, 800 XL oder 130 XE besitzen. Das Buch enthält eine vollständige Liste aller Systemadressen und [illegible] Speicheraufteilung und Ausnutzung des Betriebssystems, einen Tabellenteil mit Zeichencodes, Adressen, Kommandos, [illegible], Fehlermeldungen sowie einen alphabetischen Sachwortregister.

Bestellnummer SY 108 **DM 48,–**

Hal Hickerhan

Der ATARI als Musikbox

1[illegible] Seiten. Dieses Buch führt Sie in die Grundbegriffe der Musik ein und [illegible] die Möglichkeiten der Computermusik [illegible]. Das Buch ist für Anfänger geschrieben und zeigt, wie [illegible] einfachen [illegible] Akkorde und Kanons [illegible]. Sie müssen [illegible] noch ein Musikinstrument spielen können, um mit [illegible] Ihren ATARI in ein mehrstimmiges Orchester zu verwandeln.

Bestellnummer MT 123 **DM 37,00**

Chaos Computer Club

Die Hackerbibel

[illegible] Seiten
Überall in den Medien ist seit [illegible] die Rede von den Hackern. Und wenn man Hacker sagt, meint man in [illegible] Mitglieder des Chaos Computer Club Hamburg und seine Ableger. [illegible] 135.000 DM Coup [illegible] Hamburger Sparkasse oder die diversen Aufsätze [illegible] Frank Elstner und anderen. [illegible] Buch von [illegible] Nicht nur über das "Wie" des Hackens, sondern auch über das "Warum".

Bestellnummer H 800 **DM 3[illegible]**

BUCH-BESTELLKARTE

Bitte liefern Sie mir folgende Bücher

Anzahl	Bestell-Nr.	Titel	Einzel-Preis inkl. MwSt.

Name des Bestellers

Anschrift

PLZ/Ort

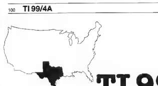
Datum

Ich wünsche folgende Bezahlung:
☐ Nachnahme ([illegible] DM Porto + Versandkosten)
☐ Vorauskasse (keine Versandkosten)

Bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe 48425-706 überweisen.

Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden an Computer Kontakt, Buchversand, Postfach [illegible], 7[illegible] [illegible]



TI 99/4A

Liebe TI-Freunde!

In dieser Ausgabe gibt es viel Neues zu berichten. Zuerst die Meldung, auf die alle TI-Besitzer schon seit Monaten warten: Der neue TI, lange Zeit als Gerücht gehandelt, steht endlich vor der Tür. Er soll nach letzten Meldungen als Einschubkarte für die Peripheriebox ausgeführt sein. Auf dieser Karte befinden sich 640 KByte RAM, die bis auf 2 MByte ausgebaut werden können. Das Herz des neuen TI schlägt mit 12 MHz und wird vom Prozessor TMS 9995 gebildet. Dieser Prozessor ist der kleine Bruder des TMS 9900, der sich im TI 99/4A befindet. Der Datenbus des TMS 9995 ist intern 16 Bit breit, während nur 8 Datenleitungen nach außen führen. Im Prozessorchip ist zusätzlich ein 256-Byte-RAM integriert, auf das der Prozessor wesentlich schneller zugreifen kann als auf externe Speicherchips. Als Bildschirm kann ein 80-Zeichen-Farbmonitor benutzt werden. Eingebaute Unterstützungsroutinen erlauben nun die Anwendung der Programme "TI-Writer" und "MS Multiplan". Eine neue IBM-XT-Tastatur ist neben einem Modulport und einer RS 232 Schnittstelle ebenfalls vorgesehen. Die Kompatibilität ist assemblermäßig zu 95 % gewährleistet, während das eingebaute Basic dem Microsoft-Basic der IBM-Rechner entspricht und somit mit dem TI-Basic nicht mehr viel gemein hat. Die Maschine soll nach Mitteilungen von Myarc ca. 500 Dollar kosten und unter dem Namen Genesis auf den Markt kommen.

Zum nächsten Punkt: Der TI 99er Workshop Rheinland möchte zu Anfang des Jahres 87 einen TI-Treff in Köln veranstalten, der alle bisherigen Veranstaltungen dieser Art überbieten soll. Viele Clubs aus dem Ausland haben ihr Kommen bereits zugesagt, die Stadt Köln und einige ortsansässige Unternehmen unterstützen die Aktion finanziell. Nur die deutschen TI-Clubs haben bislang nicht viel Interesse gezeigt. Daher möchte ich an dieser Stelle alle TI-Benutzer und Interessengemeinschaften, die Zeit und Lust haben, an einer solchen Veranstaltung teilzunehmen, auffordern, sich beim Workshop zu melden, damit die Teilnehmerzahl ungefähr abgeschätzt werden kann. Es wäre schade, wenn die Veranstaltung mangels Interesse nicht stattfinden könnte.

Doch nun zum heutigen Inhalt. Es gibt auch hier wieder eine kleine Sensation zu vermelden: Der TI wurde um eine Programmiersprache erweitert. Ab sofort kann man mit ihm auch in C sprechen. Wer eine Speichererweiterung, ein Diskettensystem und ein Editor/Assembler-Modul besitzt, kann für ganze 10,– DM einen C-Compiler als Freeware erwerben und die neue Sprache ausprobieren. Als weitere Neuigkeit haben wir ein EPROM-Programmiergerät von der Firma Mechatronic getestet. Und der Aktivität unserer Leser verdanken wir einen interessanten Artikel über neue Software für das Grafikprogramm "TI-Artist". Zwei Spiele in Extended Basic, ein Utility zum Beschleunigen von Basic-Programmen und ein Assemblerprogramm, das das Laden und Speichern von überlangen Basic-Programmen ermöglicht, runden die heutige Ausgabe ab. Ich wünsche euch wieder viel Spaß und hoffe, daß ihr auch weiterhin so aktiv bleibt wie in den vergangenen Wochen und uns eure persönlichen Erfahrungen und Ansichten mitteilt.

H. J. Schwanick

TI-Software-Hitparade

So langsam scheint die Sache ins Laufen zu kommen, aber wie gesagt, ein paar mehr können es schon noch werden.

Der Spitzenreiter ist diesmal der alte, aber auf den Plätzen dahinter tut sich einiges. Das liegt auch daran, daß immer noch neue Software aus den USA nach Deutschland kommt.

1. TI-Artist II	(1)	A, AS
2. Micro Pinball	(–)	S, AS
3. Tennis	(3)	S, AS
4. War Games	(–)	S, AS
5. DM 1000	(4)	A, AS
6. Bigfoot	(2)	S, AS
7. Artillerie	(9)	S, BA
8. Ret. t. P. Island	(10)	S, MO
9. Disk-Katalog (CK)	(6)	A, AS
10. TI-Banner	(8)	S, AS

Der neue Gewinner ist
Ralf Bauer, Kelterstr. 36, 6962 Adelsheim-Sennfeld

Er erhält ein brandneues Softwarepaket aus den USA. Ihr seht, es lohnt sich mitzumachen. Ab sofort nehmen übrigens alle Einsendungen an der Verlosung teil.

Eure Tips müßt ihr wieder an folgende Anschrift einschicken:

TI 99er Workshop Rheinland
Dept. Allgemein + Software
c/o Mike Heuser
Karl-Marx-Allee 18
5000 Köln 71

Heiße Bits für den TI mit dem Mechatronic-Eprommer 99

Von vielen TI-Freaks sehnsüchtig erwartet, gibt es jetzt ein EPROM-Programmiergerät für den TI. Mit diesem Zusatz wird der gute alte TI zur Entwicklungsmaschine für TMS-9900-Maschinenprogramme. Zusammen mit den bereits erhältlichen EPROM- und GROM-Karten kann jeder TI-Besitzer nun seine eigenen Module herstellen.

Der Eprommer befindet sich in einem stabilen Metallgehäuse und wird in den Modulport eingeschoben, ist aber doppelt so lang wie ein normales Modul. Zwei Gummifüße an der Unterseite des Gehäuses stützen den gesamten Eprommer ab, damit die mechanischen Belastungen an Modulport und Eprommerplatine nicht zu Beschädigungen führen. Um das Netzteil der Konsole nicht zu überlasten, übernimmt ein separates Steckernetzteil die Spannungsversorgung. Auf der Gehäuseoberfläche befinden sich ein 28poliger IC-Nullkraft-Sockel, ein 36poliger Steckverbinder, der zur Aufnahme eines Moduls dient, und vier LEDs, die den Betriebszustand des Eprommers signalisieren. Alles in allem macht der gesamte Aufbau einen sauberen und vertrauenerweckenden Eindruck.

Mit dem Mechatronic-Eprommer kann man insgesamt 7 verschiedene EPROM-Typen programmieren. Die Skala reicht vom Typ 2716 mit 2 KByte bis zum Speicherriesen 27128 mit einer Kapazität von 16 KByte. Da der Eprommer vollständig softwaregesteuert ist, entfällt das lästige und fehlerträchtige Einstellen von Dip-Schaltern und Programmier-Spannungen. Die Betriebssoftware kann unter den Modulen Extended Basic, Editor/Assembler oder Minimemory gestartet werden. In jedem Falle sind außerdem eine 32 KByte Speichererweiterung und ein Diskettenlaufwerk nötig.

Die Arbeit mit dem Eprommer ist sehr einfach. Zuerst steckt man ihn in den Modulport der Konsole. Danach wird das zum Betrieb der Software nötige Modul, beispielsweise das Extended Basic Modul, in die Steckerleiste des Eprommers gesteckt. Erst jetzt darf man laut Handbuch die Konsole einschalten, da ein Modulwechsel bei eingeschaltetem System den Eprommer, die Konsole oder das Modul selbst beschädigen könnte. Nun wird das Steckernetzteil mit dem Eprommer verbunden. Dann kann es endlich losgehen. Die Betriebssoftware für den Eprommer wird automatisch bei der Auswahl des Extended Basic Moduls von Diskette geladen, und nach wenigen Sekunden erscheint das Hauptmenü auf dem Bildschirm, das zwei Auswahlmöglichkeiten erlaubt: EPROMMER COMMANDS und BUFFER COMMANDS. Unter dem Punkt EPROMMER COMMANDS kann man den EPROM-Typ auswählen und danach mit dem Inhalt des RAM-Puffers programmieren. Weitere Möglichkeiten unter EPROMMER COMMANDS bestehen darin, den Inhalt eines EPROMs in den RAM-Puffer zu lesen, seinen Inhalt mit dem RAM-Puffer zu vergleichen und zu überprüfen, ob ein EPROM völlig gelöscht ist. Als Pufferspeicher dient in allen Fällen der RAM-Bereich von A000 bis DFFF. Anwenderfreundlich ist auch, daß man beliebig große Teile des EPROMs beschreiben kann und nicht gezwungen ist, für wenige Byte das ganze EPROM zu programmieren, was mit einer entsprechend langen Wartezeit verbunden wäre. In diesem Zusammenhang soll auch erwähnt werden, daß EPROMs vom Typ 2764 und 27128 im sogenannten Fast Programming Mode gebrannt werden können, was eine erhebliche Zeitersparnis zur Folge hat. Wenn der Programmiervorgang gestartet ist, erscheint in der untersten Bildschirmzeile die Adresse der in Arbeit befindlichen Speicherzelle, so daß man während der Programmierzeit wenigstens sieht, daß sich etwas tut.

Unter dem Hauptmenüpunkt BUFFER COMMANDS befinden sich Kommandos, mit denen der Pufferspeicher bearbeitet werden kann. Zunächst findet man die unterschiedlichsten Ladekommandos. Der Pufferspeicher kann aus anderen RAM-, GROM- oder VDP-Bereichen geladen werden, ebenso ist das Laden von Assembler Files von Diskette sowohl im Tagged Object Format als auch im Program Image Format möglich. Eine wertvolle Ergänzung bietet ein Hex-Editor, mit dem sich einzelne Speicherzellen manuell ändern lassen. Um mühsam erstellte Speicherinhalte zu sichern oder den Inhalt fremder EPROMs zu archivieren, kann der gesamte Pufferinhalt im Program Image Format auf Diskette gespeichert werden.

Computer-Kontakt
das Heft mit den
preisgünstigen
Kleinanzeigen

Der Mechatronic-Eprommer ist für Programmentwickler, die eigene Module herstellen wollen, genauso zu empfehlen wie für Anwender, die EPROMs auf völlig anderen Gebieten einsetzen wollen (z. B. Einplatinen-Mikrocomputer in Steuerungen). Durch die solide Ausführung und die flexible Software ist dieses Gerät auch für den professionellen Einsatz denkbar. Der Hobbyanwender muß auf jeden Fall neben den Anschaffungskosten von ca. 280 DM auch noch den Preis für eine UV-Lampe zum Löschen der EPROMs einkalkulieren.

H. P. Schwarck

USA-SPECIAL

und noch vieles mehr! Vorbestellung heute noch absenden ...!

CSGD GRAFIC

MAUS-EDITOR

SUPER ACTION

HARDWARE

TI-ARTIST SUPPORT

Preise

Da ist der Weihnachtsmann platt!

TI-99/4A Total ...

TORPEDO BASIC V. 3.0

Das stärkste Programm für Ihren TI

Features:

Komplettpreis nur DM 129,–

SUPERINFO GRATIS

Rausch & Haub
Vertriebsgesellschaft mbH
Postfach 3 01 32
5300 BONN 3

0228/333864

Ein Künstler und seine Gesellen

Neue TI Software

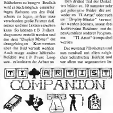
Es gibt seit einiger Zeit in Deutschland bei verschiedenen Händlern Bestrebungen, endlich wieder gute Software aus den USA zu importieren und zu vernünftigen Preisen anzubieten. Dies ist nicht zuletzt ein Verdienst des TI 99er Workshop Rheinland, der seine Partner immer wieder auf die Möglichkeiten des Softwareeinkaufs in den USA hingewiesen hat.

Seit einiger Zeit ist nun auch auf dem deutschen Markt das hervorragende Zeichen- und Grafikprogramm "TI Artist" erhältlich. Nachdem in einem anderen Magazin schon ein detaillierter Testbericht veröffentlicht wurde, hier nur noch einmal eine kurze Zusammenfassung der wichtigsten Merkmale des "TI Artist 2.0".

Es handelt sich um ein Programm, mit dem auf einfachste Weise hervorragende Grafiken auf dem Bildschirm erstellt werden können. Ebenso leicht lassen sich verschiedene Schriftarten und kleine Hilfsbilder, die man später in Großbildern zusammenfügen kann, erstellen. Sogar der Aufbau elektronischer Schaltbilder ist möglich, da die gebräuchlichsten Zeichen (z.B. für Transistoren) schon im Programm vordefiniert sind. Diese können, bevor man sie positioniert, in alle Richtungen gedreht werden. Nachdem ein Zeichen die richtige Lage hat, wird es mit dem Joystick, einer Maus oder den Cursor-Tasten in das Bild eingebaut.

Im eigentlichen Hauptmenü findet man eine Vielzahl von Funktionen, die von der einfachen DRAW-Routine über CIRCLE- und FILL-Routinen bis hin zur Auswahl der Strichstärke gehen. Soweit die Kurzbeschreibung des "TI Artist", der mittlerweile bei einigen deutschen Händlern für ca. 98 DM zu erstehen ist.

Nun aber zu seinen Gesellen. In einer amerikanischen TI-Fachzeitschrift fand sich eines Tages eine Werbeseite, auf der Zubehör für den "TI Artist" angeboten wurde. Dieses Paket besteht aus insgesamt 10 (!) Disketten voll mit Grafiken, Zusatzfiles, Hilfsroutinen und vor allem Schriftzeichen, die man ohne große Anstrengungen mit dem "TI Artist" verarbeiten oder verbinden kann. Zuerst einmal gibt es da den "Display Master". Mit diesem Programm kann man mit Hilfe des "TI Writer" oder des E/A-Editors eigene Hilfsprogramme schreiben, um Bilder, die mit dem "TI Artist" erstellt wurden, im Extended Basic auf den Bildschirm zu bringen. Endlich wird es auch möglich, einen farbigen Rahmen um den Bildschirm zu legen, indem man verschieden große Fenster definieren und mit Texten versehen kann. So können z.B. Balkendiagramme erstellt werden und mit dem "Display Master" die dazugehörigen Kommentare über das Bild verteilt werden. Auch einige andere hilfreiche Befehle wie z.B. Pause, Loop usw. erleichtern die Arbeit ungemein. Zum Lieferumfang gehören neben einer sehr ausführlichen Befehlsliste und einer Betriebsanleitung 4 Demobilder von einer Qualität, die man bisher nur von Rechnern der gehobeneren Preisklasse wie Commodore Amiga oder Atari ST kennt. Hier zeigt der TI wieder einmal, was wirklich in ihm steckt.



Neugestaltete Präsentation des TI Artist

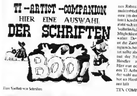
Auf zwei zusätzlichen Disketten liefert derselbe Hersteller, die Fa. Inscebot Inc., weiteres sog. "Artist Extras". Zum einen befinden sich auf diesen Disketten einige sehr schöne Schrifttypen. Weiterhin erhält man einige Unterstützungsroutinen, die direkt in den "TI Artist" eingebunden werden können, so z.B. die Input-DSR für eine Maus, die es ja inzwischen auch hier zu kaufen gibt.

Den dritten Teil der Disketten bilden ca. 10 meist sehr gut gelungene Bilder, die entweder im "TI Artist" oder auch im "Display Master" verwendet werden können, sowie drei Konversions-Routinen, mit denen Grafiken anderer Programme "TI Artist"-kompatibel werden.

Die weiteren 7 Disketten sind nun randvoll mit allen möglichen Schriftarten, Bildern und vor allem mit sogenannten Instances. Dies sind kleine Bildchen von ca. 5×5 cm, die an jede beliebige Bildschirmposition geschoben, vervielfältigt, kopiert und bearbeitet werden können. Ein weiterer Vorteil der Instances ist, daß sich Muster und Grafiken auch im Extended Basic mit einem sog. CSGD-File, das sich ebenfalls auf den "Artist Extras"-Disketten befindet, erstellen lassen. Sie können mit einem Konversions-Programm auf den "TI Artist" umgewandelt oder im Extended Basic benutzt werden. Einen besonderen Clou bieten alle Schriftarten, die sich sowohl in Normalschwarz als auch invers darstellen lassen, so daß sich ihre Anzahl noch einmal verdoppelt. Eine kleine Auswahl dieser Instances sowie einiger weniger Schriftarten aus diesem Package haben wir diesem Artikel als Hardcopy beigefügt, damit man einmal einen Eindruck von der Leistungsfähigkeit dieses Pakets bekommt.

Insgesamt bekommt der Käufer für sein Geld eine fast riesig zu nennende Menge Software, die kaum einen Wunsch in punkto Grafik offenläßt. Auf den 10 Disketten befinden sich ungefähr 200 (!) Routinen und Hilfsfiles, wobei aber die Masse nicht auf Kosten der Klasse geht. Ein weiterer Vorteil dieses Pakets ist ohne Zweifel, daß man nun nicht alle 10 Disketten auf einmal kaufen muß, sondern ganz gezielt auswählen kann, was für einen selbst nutzbringend oder erschwinglich ist. So ist es z.B. möglich, den "Display Master" oder die "Artist Extras" einzeln zu erwerben. Die 7 Companion-Disketten sind aufgeteilt in zwei Pakete zu 5 und 2 Disketten. Ebenso kann man alle Teile des Pakets miteinander kombinieren (z.B. 2 Companion-Disketten und den "Display-Master"), so daß den Möglichkeiten kaum Grenzen gesetzt sind.

Auf unser Betreiben hin hat sich die Fa. Rausch & Haub in Bonn dazu entschlossen, das gesamte Paket für die deutschen TI-User zu importieren. Die aktuellen Preise kann man der CK-Werbung oder der Mailbox des TI 99er Workshop Rheinland, Tel. 0228/658232 entnehmen.



Eine Vielfalt von Schriften

Noch ein Wort an alle diejenigen, die uns in letzter Zeit fragen, wie man denn wohl in den USA einkaufen könne. Hier einige Tips dazu. Leider gibt es keine Möglichkeit, kleine Geldbeträge an die Lieferanten zu überweisen, da in den allermeisten Fällen keine Bankverbindung bekannt ist. Auch Lieferung per Nachnahme (C.O.D.) kommt für die US Händler nicht in Frage, das Mißtrauen ist drüben allgemein sehr groß. Die meisten Händler bieten drei Wege der Bestellung und Bezahlung an. Die erste Möglichkeit besteht darin, seiner Bestellung den Kaufpreis in bar beizulegen. Man sollte aber dafür sorgen, daß es von außen nicht möglich ist, das Geld im Umschlag zu sehen oder zu ertasten. Dies kann man z.B. durch ein Stück schwarze Pappe verhindern, das man mit in den Umschlag legt. Wir haben bisher nur gute Erfahrungen mit dieser Art des Einkaufs in den USA gemacht. Für Leute, die eine Kreditkarte besitzen, gibt es nun auch noch die Möglichkeit, telefonisch bei den Händlern zu bestellen und bei der Bestellung die Nr. der Karte sowie das Gültigkeitsdatum anzugeben. Hier gewähren die meisten Händler auch noch einen kleinen Rabatt. Inwieweit der normalsterbliche TI User nun über eine (in den USA sehr verbreitete) Kredit Karte verfügt, entzieht sich unserer Kenntnis, der Vollständigkeit halber sei diese Möglichkeit hier aber auch erwähnt. Die dritte Möglichkeit ist die Zahlung per bankbestätigtem Scheck. Diese Zahlungsart sollte aber vor dem Einkauf mit der Bank und dem US Händler abgeklärt werden. Hier nun noch die Adresse eines TI Anbieters in den USA, der wohl mit das größte Angebot an Hard- und Software bereit hält:

TEX COMP
P.O. Box 33 064
Granada Hills
CA 91344
USA

Gegen Zusendung von 2 US$ erhält man von dort einen 40seitigen Katalog mit allem, was des TI Users Herz begehrt.

Das war's vorerst wieder einmal vom TI 99er Workshop Rheinland. Wer Fragen bezüglich des "TI Artist"-Companion Packages hat, wende sich entweder an die bekannte Adresse oder direkt an Rausch & Haub GdbR, 5300 Bonn.

TI 99er Workshop Rheinland
Mike Heuser

Merry Christmas

Computer Kontakt schließt sich diesem Gruß an

Neues vom Turbo-Assembler

Der Turbo-Assembler hat sich zum Bestseller unter unseren angebotenen Programmen gemausert. Aufgrund der hervorragenden Eigenschaften dieses Programms erscheint dies auch nicht weiter erstaunlich. Immerhin arbeitet der Turbo-Assembler mindestens doppelt so schnell wie der TI Assembler, der Bedarf an Speicherplatz auf der Diskette von Quell- und Objectcode ist bis zu 3fach geringer, während die Zeit zum Laden der fertigen Programme 5mal kürzer ist. Dies sind schon sehr überzeugende Vorteile. Doch damit nicht genug. Der Turbo-Assembler ist der einzige Assembler für den TI, der auch ohne Diskettenlaufwerk auskommt und notfalls auch mit Cassettenrecorder sehr gute Ergebnisse erzielt. Da aber gute Dinge auch noch verbessert werden können, wurden einige Schönheitsoperationen am Turbo-Assembler vorgenommen.

Aufgrund der konstruktiven Zusammenarbeit mit den bisherigen Benutzern wurde das Handbuch neu überarbeitet. Viele Tips und Anregungen flossen dabei in das neue Konzept mit ein, so daß der Umfang auf insgesamt 19 Seiten angewachsen ist. Das auf der Diskette enthaltene Katalogprogramm, das auch in der CK 6/85 als Listing des Monats erschien, wurde um einige wertvolle Optionen erweitert. Der Speicherplatz für die Einträge wuchs auf stattliche 30 KByte an, so daß auch sehr umfangreiche Programmsammlungen erfaßt werden können. Eine von den Anwendern stürmisch geforderte Funktion wurde nun ebenfalls verwirklicht. Der gesamte Kataloginhalt kann jetzt auch als Datei auf einer Diskette gespeichert werden. Damit lassen sich Updates nun noch schneller durchführen. Wer bislang Schwierigkeiten mit der Druckersteuerung hatte, braucht ebenfalls nicht mehr zu verzweifeln. Eine zusätzliche Druckroutine bedient auch Drucker, die nicht dem Epson Standard entsprechen. Neben diesen wesentlichen Erweiterungen kann man die permanent zur Verfügung stehenden Kleinbuchstaben schon fast als Spielerei ansehen. Dies alles ist – wie schon gesagt – im neuen Turbo Assembler Paket enthalten.

Wer schon einen Turbo Assembler besitzt, der kann seine Originaldiskette einschicken. Gegen einen Unkostenbeitrag von 5,– DM erhält er die neue Katalogversion und zusätzlich das erweiterte Handbuch. Wer keinen Turbo-Assembler hat, kann eine Freeware Diskette mit dem neuen Katalog und einem Lader Utility direkt bei H.P. Schwaneck, Roggenkamp 3, 3300 Braunschweig anfordern.

Lange Programme auf dem TI 99 speichern

Das Extended Basic des TI 99 hat einen Nachteil: Bei angeschlossener Speichererweiterung können zwar Basic-Programme bis zu einer Länge von 24 KByte geladen werden, die aber dann nicht mehr auf Cassette, sondern nur noch auf Diskette gespeichert werden können. Das später folgende Maschinenprogramm für Extended Basic und 32-K-Erweiterung bietet Ihnen zwei Befehle zur Speicherung solcher Programme auf Cassette.

Will man den Grund für diese Eigenart des Basic verstehen, muß man einen tieferen Blick ins System werfen. In der Grundversion des TI 99 sind in der Konsole nur 16 KByte Speicher eingebaut, die vollständig unter Kontrolle des Video-Chips stehen. Erst die 32-K-Erweiterung schafft einen echten, direkt adressierbaren RAM-Bereich, in dem Basic-Programme gespeichert werden.

Die Cassetten-Routinen haben nun eine Eigenart: Daten von oder zur Cassette müssen über das Video-RAM geschickt werden. Soll ein Programm im kompakten PROGRAM-Format gespeichert werden, muß es vollständig als ein Block im Video-RAM liegen. Da dieses aber nur 16 KByte groß ist und die eigentlichen Video-Tabellen und Diskettenpuffer davon noch abgezogen werden müssen, dürfen Programme maximal rund 13 KByte haben, wenn sie auf Cassette gespeichert werden sollen.

Dieses Problem läßt sich nur in Maschinensprache lösen, indem das Basic-Programm in zwei nacheinander zu speichernde Blöcke aufgeteilt wird. Dazu haben Sie zwei Befehle: CALL LINK ("SVLONG") speichert ein überlanges Programm auf Cassette ab, CALL LINK ("LDLONG") lädt es wieder. Dabei erscheinen zweimal die normalen Meldungen wie bei OLD und SAVE. Natürlich dürfen Sie nach dem Speichern oder Laden des ersten Blocks das Band nicht zurückspulen, auch wenn der Computer Sie dazu auffordert. Bei kurzen Basic-Programmen, die sich normal speichern lassen, gibt das Programm die Fehlermeldung "Bad Value" aus, da es wegen eines möglichen erheblichen Zeitverlustes nicht sinnvoll ist, dieses Programm zu verwenden.

Am besten gehen Sie beim Speichern so vor:

- Das Maschinenprogramm eintippen oder laden
- Das zu speichernde Basic-Programm einladen
- SAVE CS1 eingeben
- Bei Fehlermeldung CALL FILES (1) eingeben und eine nicht vorhandene Zeile löschen; das bringt rund 1 KByte mehr Platz im Video-RAM
- Erneut SAVE CS1 eingeben
- Erst wenn jetzt wieder eine Fehlermeldung kommt, CALL LINK ("SVLONG") eingeben

Das hört sich zwar recht kompliziert an, ist aber ganz logisch. Erheblich einfacher ist nun das Einlesen eines so gespeicherten Programms: CALL LINK ("LDLONG") eingeben und das Band starten. Sobald der erste Programmblock geladen ist, müssen Sie entsprechend den Meldungen auf dem Bildschirm den zweiten Block einlesen.

Noch ein Hinweis: Da das Maschinenprogramm im Low Memory liegt und hier für das Video-RAM einen 6-KByte-Buffer benötigt, dürfen solange keine anderen Maschinenprogramme mehr in diesen Bereich geladen werden, wie man die Cassetten-Routinen verwendet.

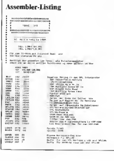
Marius Kotulla

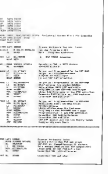
MC-Generator

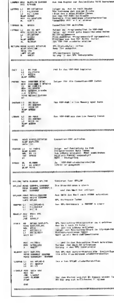
```
10 '*************************
20 '*    POKELISTE FUER     *
30 '*    LOAD/SAVE LONG     *
40 '* ERSTELLT AM 9.9.86    *
50 '*   VON H-P SCHWANECK   *
60 '*************************
100 DIM CH(40):: CALL CLEAR :: PRINT "ICH UEBERPRUEFE DIE DATA-   ZEILEN. BITTE
WARTEN"
110 RESTORE 810 :: FOR I=1 TO 39 :: READ CH(I):: NEXT I
120 RESTORE 420 :: FOR Z=1 TO 39 :: SUM=0 :: FOR I=1 TO 16 :: READ A :: SUM=SUM+
A :: NEXT I :: IF SUM<>CH(Z)THEN PRINT "EINGABEFEHLER IN ZEILE";Z*10+410 :: CALL
 SOUND(100,110,0):: STOP
130 NEXT Z
140 PRINT "EINGABE KORREKT, JETZT WIRD GEPOKED"
150 RESTORE 420 :: ADR=9460 :: CALL INIT
160 READ A :: IF A<>-1 THEN CALL LOAD(ADR,A):: ADR=ADR+1 :: GOTO 160
400 CALL LOAD(8194,36,244,63,240)
410 CALL LOAD(16368,76,68,76,79,78,71,37,222,83,86,76,79,78,71,37,112)
420 DATA 005,006,000,000,000,000,000,000,000,000,000,000,203,020,203,053
430 DATA 203,078,203,231,204,071,204,150,204,228,205,029,205,075,205,096
440 DATA 033,131,035,253,030,184,040,183,041,182,042,195,043,193,044,173
450 DATA 045,194,047,196,058,181,059,180,060,191,061,190,062,192,094,197
460 DATA 255,058,058,130,065,084,240,071,079,133,073,070,132,079,078,155
470 DATA 075,082,185,080,073,221,084,079,177,255,065,066,083,203,065,076
480 DATA 000,000,000,003,004,000,008,000,000,058,207,208,006,012,009,196
490 DATA 080,080,000,000,096,003,067,080,049,000,255,255,002,224,036,246
500 DATA 136,032,131,048,037,094,026,004,002,000,030,000,004,032,032,052
510 DATA 216,032,036,245,037,096,006,160,038,146,002,000,009,196,002,001
520 DATA 131,048,002,002,000,004,004,032,032,036,002,000,009,200,002,001
```

```
530 DATA 208,000,002,002,048,000,004,032,032,036,200,002,037,102,006,160
540 DATA 038,092,006,160,038,182,002,000,009,196,192,096,131,048,002,002
550 DATA 207,255,096,129,004,032,032,036,200,002,037,102,006,160,038,092
560 DATA 006,160,038,182,006,160,038,164,016,053,002,224,036,246,216,032
570 DATA 036,244,037,056,006,160,038,146,002,002,048,000,200,002,037,102
580 DATA 006,160,038,092,006,160,038,182,002,000,009,196,002,001,131,048
590 DATA 002,002,000,004,004,032,032,044,200,032,131,048,131,134,006,032
600 DATA 131,134,002,000,009,200,002,001,208,000,002,002,048,000,004,032
610 DATA 032,044,200,002,037,102,006,160,038,092,006,160,038,182,002,000
620 DATA 009,196,192,096,131,048,002,002,207,255,096,129,004,032,032,044
630 DATA 006,160,038,164,216,032,037,084,131,124,004,032,038,220,000,052
640 DATA 002,224,131,224,004,096,000,112,002,000,003,209,002,001,037,096
650 DATA 002,002,000,018,004,032,032,036,200,032,037,106,131,074,200,032
660 DATA 037,100,131,076,002,000,003,221,200,000,131,066,200,032,037,086
670 DATA 131,084,004,224,131,208,216,032,037,090,131,109,004,091,002,000
680 DATA 033,242,002,001,039,216,002,002,023,212,004,032,032,044,004,091
690 DATA 002,000,033,242,002,001,035,216,002,002,023,212,004,032,032,036
700 DATA 004,091,216,032,037,084,131,124,004,032,038,220,000,061,002,000
710 DATA 003,209,004,032,032,040,009,209,022,001,004,091,006,160,038,164
720 DATA 002,000,036,000,004,032,032,052,037,022,008,224,216,032,152,002
730 DATA 037,082,216,032,152,002,037,083,006,032,037,082,216,062,156,002
740 DATA 216,062,156,002,002,224,131,224,002,000,131,200,002,001,037,054
750 DATA 002,002,000,013,204,112,006,002,022,253,184,032,037,088,131,115
760 DATA 208,160,131,115,009,130,002,034,130,254,002,000,039,046,006,064
770 DATA 204,128,196,160,037,092,004,096,000,112,006,096,131,115,216,032
780 DATA 037,082,156,002,216,032,037,083,156,002,002,000,131,230,002,001
790 DATA 037,054,002,002,000,013,204,049,006,002,022,253,002,224,037,022
800 DATA 003,128,0,0,0,0,0,0,0,0,0,0,0,0,0,-1
805 'PRUEFSUMMEN
810 DATA 490,2591,1816,2007,1760,1874,711,1476,660,1222,505,871,1194,1428,1579,1
156
820 DATA 1071,834,775,1101,1475,1338,1142,938,1350,1494,979,878,1076,1024,881,12
34
830 DATA 1474,1203,1330,1625,1169,929,130
```

Assembler-Listing







C auf dem TI 99/4A

Der gute alte TI wird langsam zum Sprachenkünstler. Nach Assembler, Basic, Pascal und Forth ist nun seit kurzer Zeit auch die Programmiersprache C verfügbar. C ermöglicht das strukturierte, modulare Programmieren wie in Pascal. C-Programme laufen jedoch um einiges schneller als vergleichbare Pascal-Programme. Selbst gegenüber dem in der Anwendung eher etwas spröden Forth ist noch ein geringer Geschwindigkeitsvorteil vorhanden.

Der C-Compiler für den TI wurde von dem Amerikaner Clint Pulley entwickelt und liegt mittlerweile in einer zweiten verbesserten Version vor. Erstaunlicherweise genügte zur Entwicklung ein normaler TI 99 mit 32 K-Speichererweiterung, RS 232-Schnittstellenkarte und ein Diskettensystem mit 2 Laufwerken – ein weiterer Beweis für die Leistungsfähigkeit unserer texanischen Rechen- und Spielmaschine. Der C-Compiler entspricht den Spezifikationen des Standard C nach Kernighan und Jeff Lomicka und läuft auf dem TI mit allen Erweiterungen, die für den Betrieb des Editor/Assembler Moduls nötig sind. Dies wären 32 KByte Speichererweiterung, Diskettensystem und selbstverständlich das Editor/Assembler Modul selbst. Empfehlenswert, aber nicht unbedingt nötig, ist ein Drucker samt Schnittstellenkarte. Der C-Compiler versteht ein abgemagertes Standard C, das um viele nützliche Unterprogramm-Bibliotheken mit TI 99-spezifischen Befehlen ergänzt wird. Die Arbeitsweise entspricht dem One Pass System, d. h. der Programmcode wird in einem Durchgang auf Fehler geprüft und übersetzt, was den Nachteil hat, daß alle Referenzen zur Compilationszeit definiert sein müssen. Das bei der Übersetzung entstehende Produkt ist noch kein lauffähiger Objectcode, sondern Assembler-Quellcode im DIS/VAR 80-Format. Der lauffähige Objectcode entsteht dann im Assemblerdurchlauf.

Die Arbeit mit dem Compiler ist relativ einfach. Zunächst wird der C-Quelltext mit dem Editor des Assembler Moduls oder dem TI Writer eingegeben. Wichtig ist in diesem Zusammenhang nur, daß die erzeugte Textdatei ohne Steuerzeichen (Tabulatoren o. ä.) im DIS/VAR 80-Format auf der Diskette abgelegt wird. Nach Erstellung des Quelltextes wird der Compiler über die "Run Programm File"-Option des Assembler Moduls geladen und gestartet. Nach Angabe des Dateinamens für die Quelldatei bietet sich für den Anwender eine interessante Option: Der Compiler fragt nun, ob zum erzeugten Assembler-Quellcode die entsprechende C-Programmzeile als Kommentar hinzugefügt werden soll. Wird diese Option gewählt, dann kann der Programmierer die Auswirkung jeder Programmzeile überprüfen und den entsprechenden Assemblercode nach Bedarf optimieren. Die Ausgabe des Assemblercodes kann sowohl auf einer Diskettendatei als auch auf dem Bildschirm erfolgen. Letzteres ist wesentlich schneller und empfiehlt sich vor allem bei den ersten Durchläufen, wenn noch Tipp-Fehler im Quellcode vorhanden sind.

Sobald ein Fehler während des Compilierens auftritt, stoppt der Compiler und die entsprechende Zeilennummer wird auf dem Bildschirm angezeigt. Danach geht es mit Tastendruck weiter. Die Arbeit des Compilers kann jederzeit durch FCTN 4 gestoppt werden, so daß bei kleineren Fehlern am Anfang eines Programms nicht der gesamte Compilationsprozeß bis zum bitteren Ende abgewartet werden muß. Nach der erfolgreichen Compilation wird der Assembler aufgerufen und das Ausgabefile des Compilers zum lauffähigen Objectcode assembliert. Das fertige Programm kann mit der Editor/Assembler Option "Load and Run" geladen und gestartet werden.

Falls Bibliotheken verwendet wurden, müssen diese vor dem Programmstart ebenfalls geladen werden. Als Start Label dient immer das Wort "Start".

Auf den beiden Disketten des Compilers befinden sich außer der eigentlichen Grundausstattung noch viele nützliche Bibliotheken und Utilities. Neben den TI 99 spezifischen Ein- und Ausgaberoutinen und Dateiverwaltungsroutinen wurden beispielsweise auch die Grafikmöglichkeiten des Extended Basic adaptiert. Durch diese Routinen kann man zwischen Grafikmode und 40-Zeichenmode umschalten, Character definieren, Bildschirmfarben setzen sowie Joysticks und Keyboard abfragen. Auch eine der stärksten TI-Waffen, die Sprites, wurden nicht vergessen. Man kann Sprites erzeugen, bewegen, deren Position abfragen und vieles mehr. Die Grafikmöglichkeiten des Extended Basic werden jedoch noch übertroffen, wenn man mit der Bibliothek der Bit Map Grafikbefehle arbeitet. Diese bietet zunächst die grundlegenden Befehle, um Pixel zu setzen sowie Linien und Rechtecke zu zeichnen. Eine zweite Befehlsgruppe ist zum Mischen von Text und Bit Map Grafik vorgesehen.

Den Nachteil, daß C99 nur die Integer-Rechnung beherrscht, gleicht die Bibliothek mit Floating Point Routinen wieder aus. Man sollte diese Berechnungen jedoch nur anwenden, wenn es unbedingt nötig ist, da sie den Programmfluß stark verlangsamen. Die Besitzer eines Sprachsynthesizers können auf einen umfangreichen Wortschatz der Bibliothek "Speech" zurückgreifen. Wem die mitgelieferten Bibliotheken nicht ausreichen, der kann fehlende Befehle selbst erzeugen, und zwar sowohl in Assembler als auch in C. Kleinere Assemblerprogramme lassen sich direkt in C-Quellcode einfügen.

Ein Handbuch im üblichen Sinne gibt es allerdings nicht. Die wichtigsten Anmerkungen, Bedienungshinweise und Beschreibungen sind auf mehreren Textfiles enthalten, die man zur besseren Übersicht ausdrucken sollte (ca. 40 Seiten). Wer allerdings noch nie in C programmiert hat, der muß sich auf jeden Fall ein entsprechendes Lehrbuch kaufen, um die ersten Schritte zu erlernen. Der Compiler ist als Freeware erhältlich, d.h. man muß zunächst nur die Auslagen für Disketten und Arbeitsaufwand bezahlen. Wer später der Meinung ist, der Compiler sei brauchbar, wird gebeten, dem Autor freiwillig 20 Dollar zu übermitteln. Dafür wird der Name des Käufers registriert und er erhält immer die neuesten Updates. Wer also einen C-Compiler haben möchte, der kann diesen gegen einen frankierten Rückumschlag und 10-DM Unkostenbeitrag bei mir beziehen.

H. P. Schwaneck
[illegible] 5
3300 Braunschweig

Hexmonitor in C am TI 99/4A

Seit im Frühjahr 1986 in Europa die Existenz eines C-Compilers aus Kanada bekannt wurde und dessen Autor Clint Pulley ihn freundlicherweise als Public Domain Software zur Verfügung stellte, hat sich langsam ein kleiner Anwenderkreis dieser Pascal-Tochter gebildet. Das vorliegende Programm soll auf die Existenz von C hinweisen, das am TI ohne weitere Hardware jedem Editor-Assembler-Besitzer zur Verfügung steht.

Enthielt Version 1.0 des Compilers noch einige "Bugs", kann man dies von der Version 2.0, die zudem mit Floating point Routinen aufwartet, kaum mehr behaupten. Wieder einmal hat der alte TI den Anschluß an die derzeitige Computerwelt gefunden!

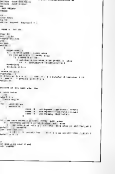
"Hexmonitor" gestattet das Screen-orientierte Editieren der Speicherinhalte in RAM, VDP und eventuellen GRAMs. Jeweils 64 Byte werden aufbereitet und können auch mit einfachem Tastendruck zu Papier gebracht werden. Dank der prozeduralen Struktur von C ist jeder Teil des rekursiv aufgebauten Programms nahezu selbstdokumentierend. Es zeigt im Ablauf eine Titelseite und nach Betätigung einer beliebigen Taste eine Kommando-Menüleiste. Nun wird die Eingabe einer vierstelligen Hexadresse erwartet: R als Kennung für ROM/RAM, V für VDP und G für GROM/GRAM-Bereich. Innerhalb des Screens kann frei editiert werden; jedes Kommando wie Vor-, Rückwärtsblättern, neue Adresse oder Ausdruck schreibt den Bildschirminhalt in den betreffenden RAM-Teil zurück.

Die Anpassung des Programms an andere Ausgabemedien als PIO wird in der Prozedur "hardcopy" durch Austausch der Device in fopen ("Devicename", "w") erzielt. Die C-Syntax entspricht dem Standard, der von Kernighan/Ritchie vorgegeben ist. Es existieren zwar Einschränkungen gegenüber einem großen C-System, diese sind aber tolerabel. Grafik-, Sound- und Sprachmöglichkeiten des TI sind durch Basic-analoge Prozeduren voll erschlossen.

Die Arbeitsgeschwindigkeit der erzeugten Programme liegt trotz Programmierkomfort einer echten Hochsprache zwischen Forth und Maschinensprache. Wie in Forth, GPL oder Maschinensprache verzögern Floatingpoint-Routinen aber den Programmablauf.

Karl Hagerbuchner

Hexmonitor



```
[illegible]
```

```
[illegible]
```

DSK 2. MONITOR; C

```
[illegible]
```

albs IMMER NEUESTE HARDWARE FÜR DEN TI 99/4A DIREKT VOM HERSTELLER

EXTENDED BASIC (Mechatronic) mit deutschem Handbuch 198,–
EXTENDED BASIC II-PLUS mit deutschem Handbuch 285,–
Extended Basic + Grafik Extended Basic (Apesoft) in 1 Modul

Umtauschaktion
Bei Bestellung eines EXTENDED BASIC II PLUS vergüten wir Ihnen DM 70,– bei kostenfreier Zusendung eines original erhaltenen schon Extended Basic Moduls (elektrisch/mechanisch einwandfreier Zustand!) Sie zahlen nur noch 215,–

Umbauaktion (gilt nur für deutschen Lizenznachbau „Mechatronic") Wir machen aus Ihrem EXTENDED-BASIC ein EXTENDED BASIC II PLUS mit deutschem Handbuch für nur 98,–

32 k RAM-ERWEITERUNG mit Centronic Interface. Kunststoffgehäuse 190 × 110 × 60 mm zum seitlichen Anstecken an den Bus, der Bus wird nach rechts durchgeschleift, mit 5 V Steckernetzteil 298,50

Unser Paketpreis-Angebot
EXTENDED BASIC II PLUS + 32 K RAM-ERWEITERUNG ohne Centronic Interface für nur 498,50

128 k RAM-ERWEITERUNG mit Centronic Interface und 5 V Steckernetzteil 389,50

Technische Änderungen vorbehalten
Versand gegen Nachnahme oder Vorauskasse

4-FARBEN PRINTER PLOTTER A 4 Centronic-Schnittstelle DIN A 4 Format Direktanschluß an 32 k- oder 128-k RAM Erweiterung 699,–
ANSCHLUSSKABEL von 32 k- oder 128-k RAM an PP A 4 98,–
SLIM-LINE LAUFWERK 5 1/4", 500-k Byte-DS/DD (z. B. TEAC FT 55 B) 398,90
EINBAUSATZ für 2 Laufwerke in original TI-P-Box 115,–
DISC STEUERKARTE (CorComp) DS/DD für max. 4 Laufwerke 635,–
Die Weltneuheit: GRAM-Karte für die P-Box mit 128k eröffnet völlig neue Aspekte von u.a. Entwicklungssystem für eigene Module 999,–
TI-MAUS – die schnelle und komfortable Cursorsteuerung mit Software auf 5,25" Diskette mit 5-V-Steckernetzteil 298,–
EPROMMER – zum Brennen von EProms bis 27128, alle Funktionen Software gesteuert, Programm auf 5 1/4 Zoll Diskette mit Steckernetzteil 278,90
TI 99/4A Intern von Heiner Martin (englische Ausgabe) 34,95

albs-Alltronic B. Schmidt Postfach 1130
7136 Ötisheim T. 07041/2747 Tx 7 263 738 albs

Neu: Die externe 80-Zeichen-Karte ist da!

SR002

In diesem Spiel für Konsole und Extended Basic geht es um die Segonen. Sie haben ein Raumgitter gebaut, mit dem sie der Galaxis die Energie entziehen, was deren Untergang bedeuten würde. Die Regierung der Vereinten Planeten rüstet Sie, den Spieler, mit dem besten Kampfschiff aus und erteilt Ihnen den Auftrag, das Raumgitter zu vernichten. Das Schicksal der Galaxis liegt nun in Ihren Händen.

Das Raumschiff ist mit Gravobomben bewaffnet, die mit einem Zeitzünder ausgestattet sind. Jetzt müssen Sie die Bomben so plazieren, daß sie über den Segonenschiffen explodieren. Zur Zerstörung einer Gitterebene benötigen Sie eine Elementarbombe der Segonen. Dazu ist das gelbe Segonenschiff zu vernichten. An seiner Stelle erscheint dann eine Elementarbombe, die Sie auffangen müssen. Nun bringen Sie das Zielkreuz über einen der Energiekonverter und feuern die Bombe ab. Sind die Konverter zerstört, gelangt man zur nächsten Ebene.

Gesteuert wird mit Joystick 1, wobei der Start mittels Feuertaste erfolgt. Im Verlauf des Spiels löst die Feuertaste dann die Gravobomben, Joystick unten die Elementarbomben aus.

Roland Tönen

```
100 CALL CLEAR :: CALL SCREEN(2)
110 CALL CHAR(58,"7E7E007C7E067E7C",59,"7C7E067C7C686C66",60,"7C7E06666667E3C")
120 FOR I=1 TO 6
130 CALL COLOR(I,3,1)
140 NEXT I
150 DISPLAY AT(11,12):"POLAND'S"
160 DISPLAY AT(14,13):";:;<<"
170 CALL MAGNIFY(3):: CALL SCREEN(2)
180 FOR I=40 TO 139
190 READ A$
200 CALL CHAR(I,A$):: NEXT I
210 DATA 7E7E007C7E067E7C,7E7E0060606077E3C,7C7E066666667E3C,7C7E067C7C686C66,7E7
E00707C607E7E,00,0,0,1C3E3A3A3A3A3E1C,0E1E3E3606060606
220 DATA 1C3E26060C1B3E3E,1C3E260E0E263E1C,0E1A3A3A3E060606,1E20203F0636361C,1C2
2203C2E2E3E1C,1C3E320206060606,1C3A3A1C3A3A3A1C
230 DATA 1C3E3A3A1E022A1C,0,0,0,0,0,0,8888888888888888,8888888888888888,1111111
1111111,1111111111111111,0F1F3F7FFFFFFFFF
240 DATA FFFFFFFFF0E0C080,F0F8FCFEFFFFFFFF,FFFFFFFF0F070301,2424242424242424,0,0
,0,0F1828486888B888,8A8B8B8F90A0C080,F018D452111D1D111
250 DATA 51D111F10D050301,0101010303030307,0F1F3F7FFFFFF3C2,8080E0C0C0C0C0E0,F0F
8FCFEFFFFCF43,0000000001010103,070F1F3F7F7F36000
260 DATA 00000000808080C0,E0F0F8FCFE3E0600,0000000000000001,03070F1F3D300000,000
00000000000080,C0E0F0F8FC0C0000,0000000001010101
270 DATA 0202,00,8080,003D7C6160610054,5400616061 7C3D00,008C3E8606860002A,2A00860
6863EEC00,1C3477F0F7B7D7E7,F7B6D6E7F077341C
280 DATA 382CEE0FEFEDEBE7,EF6D6BE70FEE2C38,040A1B3B7A5B6A7B,7B7B5A6A3B190A06,205
0D8DC5EDA56DE,DEDE5A56DCD85060
290 DATA 16367FE8A8CAEBEA,ADCAEF6C3C1C0808,686CFE17D5D3D757,D715F3363C381010,1C3
474F7B7B7B7B7,567637341C180808,382C2EEFEDEDEDEB
300 DATA 6A6EEC2C38181010,0000000103060D0F,0707070303010101,00000080C0C0508060F0,E0E040C
08080,7E7E007C7E067E7C,7C7E067C7C686C66,7C7E066666667E3C,0
310 DATA 3C7EFFFFFFFFC381,0,0,0,0040208C1F1E0F0F,0F0F1D1E8B200000,0000000C1BFBF0
E0,00F0F8F830000200,0000101C0E070707,080F1F1F3C304000
320 DATA 00020C3CF878F0F0,E0B0F0F83E0C0000,00302C22110B0B06,14102020434C80C0,030
D32C204840808,102808040C4320E00
330 FOR I=1 TO 12
340 PRINT "      @B  H          H  @B" :: PRINT "      AC  H          H  AC" :: NEXT
 I
350 DISPLAY AT(12,13)SIZE(5):"xxyzz"
360 DISPLAY AT(1,13)SIZE(5):"()*+," :: DISPLAY AT(3,12)SIZE(7):"0000000" :: DISP
LAY AT(5,14)SIZE(3):"000"
370 FOR I=1 TO 40
380 A=INT((24-1+1)*RND)+1 :: B=INT((32-1+1)*RND)+1 :: CALL GCHAR(A,B,C):: IF C>3
3 THEN 380 :: CALL HCHAR(A,B,45)
390 NEXT I
400 CALL COLOR(5,15,5):: CALL COLOR(6,15,7):: CALL COLOR(2,11,1):: CALL COLOR(3,
14,1):: CALL COLOR(4,14,1):: GOSUB 1470
410 GOTO 1410
420 CALL MUSIC :: CALL SPRITE(#1,80,16,160,128):: CALL SPRITE(#8,96,1,90,128)::
CALL SPRITE(#2,100,8,1,128,20,-2)
430 CALL COLOR(5,15,5):: CALL COLOR(6,15,7):: DISPLAY AT(12,13)SIZE(5):"     "
440 LJ=1 :: PU=10000000 :: PU=0 :: SI=1005 :: D=0 :: E=0 :: PU1=0 :: WO=128 :: R
1=0 :: R2=0 :: P3=0 :: P6=40 :: R7=80 :: R8=10
```

```
450 CALL PUNKTE(PU):: CALL SCHIFF(SI)
460 REM
470 R1=R1+1 :: R2=R2+1 :: R3=R3+1
480 IF R1=R6 THEN 490 :: IF R3=R8 THEN 520 :: IF R2>R7 THEN 500 :: GOTO 530
490 R1=R1+5 :: CALL SPRITE(#3,108,8,1,W0,20,2):: GOTO 480
500 IF LJ=1 THEN 510 :: CALL POSITION(#4,L1,L2):: IF L1>=190 AND L1<=230 THEN 51
0 ELSE 530
510 LJ=0 :: CALL DELSPRITE(#4):: CALL POSITION(#1,L3,L4):: CALL SPRITE(#4,112,8,
1,L4,20,0):: GOTO 530
520 R3=R3+5 :: W2=INT((154-99+1)*RND)+99 :: CALL SPRITE(#5,104,11,1,W2,15,0):: G
OTO 480
530 DO=10 :: CALL KEY(1,K,S):: IF K=18 THEN 620
540 DE=10 :: CALL JOYST(1,X,Y):: CALL MOTION(#1,0,X*4):: CALL MOTION(#8,0,X*4)::
 IF Y=-4 THEN 610
550 CALL COINC(#1,#2,14,T1):: CALL COINC(#1,#3,14,T2)
560 CALL COINC(#1,#4,14,T3):: CALL COINC(#1,#5,14,T4)
570 IF T1=-1 OR T2=-1 THEN 710 :: IF T3=-1 THEN 710 :: IF T4=-1 THEN 800
580 CALL POSITION(#1,P1,P2):: CALL LOCATE(#8,90,P2)
590 DO=0 :: CALL KEY(1,K,S):: IF K=18 THEN 620
600 DE=0 :: CALL JOYST(1,X,Y):: CALL MOTION(#1,0,X*4):: CALL MOTION(#8,0,X*4)::
IF Y=-4 THEN 610 ELSE 460
610 IF O=1 THEN 830 :: IF DE=10 THEN 550 :: IF DE=0 THEN 460
620 CALL POSITION(#1,A,B):: CALL SPRITE(#9,92,15,A,B,-100,0)
630 CALL SOUND(800,-7,4):: CALL COINC(#1,#2,14,T1):: CALL COINC(#1,#3,14,T2):: C
ALL COINC(#1,#4,14,T3):: IF T1=-1 THEN 710
640 IF T2=-1 OR T3=-1 THEN 710 :: CO=8 :: CALL MOTION(#9,15,0)
650 FOR I=128 TO 140 STEP 4
660 CO=CO+1 :: CALL PATTERN(#9,I):: CALL COLOR(#9,CO):: NEXT I
670 CALL JOYST(1,X,Y):: CALL MOTION(#1,0,X*4):: CALL MOTION(#8,0,X*4)
680 CALL COINC(#9,#2,10,T1):: CALL COINC(#9,#3,10,T2):: CALL COINC(#9,#4,10,T3):
: CALL COINC(#9,#5,10,T4)
690 IF T1=-1 THEN 1190 :: IF T2=-1 THEN 1250 :: IF T3=-1 THEN 1300 :: IF T4=-1 T
HEN 1360
700 IF DO=10 THEN 550 :: GOTO 600
710 SI=SI-1 :: CO=4 :: CALL DELSPRITE(#5,#5):: R3=0 :: O=0 :: CALL COLOR(#8,1)
720 FOR Q=1 TO 3
730 FOR I=128 TO 140 STEP 4
740 CO=CO+2 :: CALL PATTERN(#1,I):: CALL COLOR(#1,CO):: CALL SOUND(70,-7,4)
750 NEXT I :: NEXT Q
760 CALL DELSPRITE(#1)
770 CALL SCHIFF(SI):: IF SI<=1000.5 THEN 1410
780 FOR I=1 TO 300 :: NEXT I
790 CALL SPRITE(#1,80,15,160,128):: GOTO 550
800 IF WX=10 THEN 810 ELSE 710
810 WX=0 :: CALL DELSPRITE(#5)
820 O=1 :: CALL COLOR(#8,11):: GOTO 550
830 R3=0 :: CALL MOTION(#8,15,0):: CALL COINC(#8,#15,10,T1):: CALL COINC(#8,#16,
10,T2)
840 IF T1=-1 THEN 850 :: IF T2=-1 THEN 900 :: O=0 :: CALL COLOR(#8,1):: GOTO 550
850 FOR I=2 TO 14
860 CALL COLOR(#15,I):: CALL SOUND(40,-5,4):: NEXT I
870 CALL PATTERN(#15,68):: CALL COLOR(#15,2):: CALL COLOR(#8,1)
880 PU=PU+175 :: CALL PUNKTE(PU)
890 O=0 :: E=E+1 :: IF E>=2 THEN 950 ELSE 550
900 FOR I=1 TO 16
910 CALL COLOR(#16,I):: CALL SOUND(50,-5,4):: NEXT I
920 CALL PATTERN(#16,68):: CALL COLOR(#16,2):: CALL COLOR(#8,1)
930 PU=PU+175 :: CALL PUNKTE(PU)
940 O=0 :: E=E+1 :: IF E>=2 THEN 950 ELSE 550
950 LJ=1 :: R1=0 :: R2=0 :: R3=0 :: E=0 :: RU=RU+1 :: IF RU>=5 THEN RU=1
960 R6=R6-5 :: IF R6<=10 THEN R6=10
970 R7=R7-5 :: IF R7<=15 THEN R7=15
980 R8=R8+2 :: IF R8>=60 THEN R8=60
990 CALL DELSPRITE(#2,#3,#4,#5)
1000 CALL POSITION(#1,P1,P2):: IF P2<=125 THEN 1010 :: IF P2>=129 THEN 1020 :: G
OTO 1000
1010 CALL MOTION(#1,0,4):: GOTO 1000
1020 CALL MOTION(#1,0,-4):: GOTO 1000
```

```
1030 CALL MOTION(#1,-7,0)
1040 FOR I=80 TO 91
1050 CALL PATTERN(#1,I)
1060 FOR D=1 TO 50 :: NEXT D :: NEXT I
1070 CALL DELSPRITE(#1)
1080 FOR I=3 TO 16
1090 CALL SOUND(-200,200,5,190,4,180,3):: CALL COLOR(5,15,I):: NEXT I
1100 IF RU=1 THEN 1110 :: IF RU=2 THEN 1120 :: IF RU=3 THEN 1130 :: IF RU=4 THEN
 1140
1110 CALL COLOR(5,15,7):: CALL COLOR(6,15,13):: GOTO 1150
1120 CALL COLOR(5,15,13):: CALL COLOR(6,15,14):: GOTO 1150
1130 CALL COLOR(5,15,14):: CALL COLOR(6,15,5):: GOTO 1150
1140 CALL COLOR(5,15,5):: CALL COLOR(6,15,7):: GOTO 1150
1150 CALL PATTERN(#15,76):: CALL PATTERN(#16,76)
1160 PU1=PU1+75 :: PU=PU+PU1 :: CALL PUNKTE(PU)
1170 SI=SI+0.5 :: CALL SCHIFF(SI)
1180 CALL SPRITE(#1,80,16,160,128):: CALL SPRITE(#2,100,8,1,128,20,-2):: GOTO 55
0
1190 CALL DELSPRITE(#2)
1200 PU=PU+75 :: CALL PUNKTE(PU)
1210 FOR I=10 TO 0 STEP -2
1220 CALL SOUND(50,-6,I):: NEXT I
1230 W=INT((240-15+1)*RND)+15
1240 CALL SPRITE(#2,100,8,221,W,20,-2):: GOTO 700
1250 CALL DELSPRITE(#3):: R1=0
1260 PU=PU+100 :: CALL PUNKTE(PU)
1270 FOR I=10 TO 0 STEP -2
1280 CALL SOUND(50,-6,I):: NEXT I
1290 WO=INT((240-20+1)*RND)+20 :: GOTO 700
1300 CALL DELSPRITE(#4):: R2=0
1310 LJ=1
1320 PU=PU+125 :: CALL PUNKTE(PU)
1330 FOR I=10 TO 0 STEP -2
1340 CALL SOUND(50,-6,I):: NEXT I
1350 GOTO 700
1360 WX=10
1370 PU=PU+200 :: CALL PUNKTE(PU)
1380 FOR I=10 TO 0 STEP -2
1390 CALL SOUND(70, 5,I):: NEXT I
1400 CALL PATTERN(#5,116):: GOTO 700
1410 DISPLAY AT(12,13)SIZE(5):"xxyzz"
1420 RT=3
1430 RT=RT+1 :: IF RT>=14 THEN 1420 :: CALL COLOR(12,RT,1)
1440 CALL KEY(1,K,S):: IF K=18 THEN 1450 ELSE 1430
1450 CALL DELSPRITE(#2,#3,#4,#5):: CALL PATTERN(#15,76):: CALL PATTERN(#16,76)
1460 GOTO 420
1470 CALL SPRITE(#10,68,2,200,57,15,0):: CALL SPRITE(#15,76,2,88,57,15,0)
1480 CALL SPRITE(#11,68,2,150,193,15,0):: CALL SPRITE(#16,76,2,32,193,15,0)
1490 CALL SPRITE(#12,124,2,10,89,6,0)
1500 CALL SPRITE(#13,124,2,150,169,6,0)
1510 RETURN
1520 SUB PUNKTE(PU)
1530 B$=STR$(PU):: DISPLAY AT(3,12)SIZE(7):SEG$(B$,2,7)
1540 SUBEND
1550 SUB SCHIFF(SI)
1560 C$=STR$(SI):: DISPLAY AT(5,14)SIZE(3):SEG$(C$,2,3)
1570 SUBEND
1580 SUB MUSIC
1590 RESTORE 1640
1600 FOR I=1 TO 16
1610 READ D,T
1620 CALL SOUND(25*D,T,2,T+2,7,T*0.5,5)
1630 NEXT I
1640 DATA 2,600,5,800,5,800,5,800,2,600,5,800,5,800,5,800,2,600,5,800
1650 DATA 2,600,5,800,2,600,5,800,5,800,5,800,1,1500
1660 SUBEND
```

U-Boot-Schlacht

Dieses Spiel läuft auf dem TI mit Extended Basic. Zu Beginn erscheint das Titelbild. Um zu starten, muß die ENTER-Taste betätigt werden.

Zuerst zum Aufbau des Spielfeldes:

Sternchen (CHR$(42)): Seesterne, durch die mit dem U-Boot hindurchgefahren werden kann. Abgeschossene Torpedos explodieren jedoch darin, so daß Torpedo und Seestern vernichtet werden. Grüne Mauern: Befestigungen, die nicht überfahren werden können. Beim Zusammenprall des U-Bootes mit der Mauer ist das Schiff verloren. Torpedos explodieren hier genauso wie bei den Seesternen. Gelbe Längsstriche: Schleusen, die durchfahren und abgeschossen werden können.

Während des Spiels ist zeitweise ein Signal zu hören. Gleichzeitig erscheint ein T auf der linken bzw. rechten Seite des Bildschirms, das den entsprechenden Spieler darauf hinweist, daß sein U-Boot so bald wie möglich aufgetankt werden muß. Dazu muß es in den schwarzen Tank, der sich neben dem T befindet, hineingeführt werden. Sollte dieser vom Gegner zerstört worden sein, so ist ein Auftanken nicht mehr möglich. Dies hat zur Folge, daß das U-Boot sich bald selbst zerstört. Dasselbe geschieht auch, wenn der Spieler nach dem Signal nicht rechtzeitig den Tank erreicht. Normalerweise ist dazu jedoch genug Zeit vorhanden.

Spieler 1 spielt auf der linken Seite. Seine U-Boote und Torpedos sind schwarz, die von Spieler 2 auf der rechten Seite grau. Steuertasten für Spieler 1 sind die Pfeiltasten. Taste W dient zum Abfeuern der Torpedos. Es ist übrigens immer nur ein Torpedo pro Spieler möglich. Beim zweiten Betätigen der Schußtaste wird das alte Torpedo gelöscht und ein neues gestartet. Spieler 2 steuert ihn wie Spieler 1 mit den Tasten I – nach oben, J – nach links, K – nach rechts, M – nach unten, L – Schußtaste. Jeder erfahrene Programmierer wird die Tastatursteuerung leicht auf einen Joystick umschreiben können.

Die Torpedos werden unabhängig vom Spiel auf die Höhe des Gegners gelenkt und fliegen immer auf die gegenüberliegende Seite. Treffen zwei Torpedos aufeinander, sind beide vernichtet. Der Verlust eines U-Bootes erfolgt durch Nichtauftanken, Zusammenstoß mit einem anderen U-Boot (damit sind auch die eigenen U-Boote gemeint), Zusammenstoß mit einem feindlichen Torpedo, Aufprall auf eine grüne Befestigungsmauer.

Das Spiel ist beendet, wenn ein Spieler kein U-Boot mehr besitzt.

Um den optischen Eindruck des Spiels zu verbessern, ist der gesamte Zeichensatz in Fettschrift umgewandelt worden. Vielleicht finden diese Zeilen (460-540) auch noch in anderen Spielen oder Textprogrammen ihren Platz. Zur Erhöhung der Laufgeschwindigkeit des Programms sind die meisten Programmzeilen bis zu 5 Bildschirmzeilen lang. Noch längere Zeilen hätten jedoch die Eingabe zu kompliziert gemacht. So könnten z. B. die Zeilen 120 und 130 in einer Programmzeile von ca. 8 Bildschirmzeilen zusammengefaßt werden, ohne daß sich der Computer mit LINE TOO LONG melden würde.

Frank M. [illegible]

```
100 !U-BOOT SCHLACHT
110 ON BREAK NEXT :: GOTO 460 :: DIM G(2),H(2,5),P(2),R(2),S(2),W(2):: A,B,C,K,D
,X
120 CALL CHAR :: CALL CLEAR :: CALL COINC :: CALL COLOR :: CALL DELSPRITE :: CAL
L GCHAR :: CALL HCHAR :: CALL KEY :: CALL LOCATE
130 CALL VCHAR :: CALL MOTION :: CALL PEEK :: CALL POSITION :: CALL SCREEN :: CA
LL SOUND :: CALL SPRITE :: !@P-
140 FOR A=1 TO 2 :: C=3-A :: CALL KEY(A,K,X):: IF X=0 THEN 180
150 IF K=4 THEN 250 :: R(A)=R(A)-8*((K=0)+(K=5)*(R(A)>57)):: S(A)=S(A)+8*((K=2)-
(K=3)):: IF S(A)<1 OR S(A)>250 THEN S(A)=125+124*SGN(S(A))
160 CALL GCHAR(R(A)/8+.875,S(A)/8+.875,D):: IF D>103 AND D<121 THEN 340 :: IF W(
A)<0 THEN GOSUB 430
170 CALL LOCATE(#A,R(A),S(A))
180 W(A)=W(A)-1 :: IF W(A)=0 THEN CALL HCHAR(20,A*27-24,84):: CALL SOUND(-200,66
6,3):: CALL COINC(#A,153,A*184-151,6,X):: IF X THEN GOSUB 440
190 IF W(A)=-25 THEN 340
200 IF P(A)=0 THEN 220 :: CALL POSITION(#A+2,K,X):: CALL MOTION(#A+2,2*SGN(R(C)-
K),P(A)):: CALL PEEK(-31877,D):: IF D AND 32 THEN 270
210 CALL GCHAR(K/8+.875,X/8+.875,D):: IF D>32 THEN 240
220 CALL PEEK(-31877,D):: IF D AND 32 THEN 310
230 NEXT A :: GOTO 140
240 CALL DELSPRITE(#A+2):: P(A)=0 :: CALL HCHAR(K/8+.875,X/8+.875,32):: GOTO 220
250 CALL SOUND(-1000,111,6,-3,6,333,6):: CALL GCHAR(R(A)/8+1.875,S(A)/8+.875,X):
: IF X>32 THEN 200
260 P(A)=4*SGN(1.5-A):: CALL SPRITE(#A+2,117+A,A*13-11,R(A)+8,S(A)):: GOTO 200
270 CALL COINC(#A+2,#C,6,X):: IF X=0 THEN 300 :: CALL DELSPRITE(#A+2,#C):: P(A)=
0 :: R(C)=177 :: G(C)=G(C)+1 :: IF G(C)=6 THEN 370
280 CALL HCHAR(23,H(C,G(C)),128):: S(C)=H(C,G(C))*8-7 :: CALL SPRITE(#C,123-C*3,
C*13-11,R(C),S(C)):: CALL HCHAR(20,C*27-24,104):: GOTO 220
290 ALL HCHAR(23,H(C,G(C)),128):: S(C)=H(C,G(C))*8-7 :: CALL SPRITE(#C,123-C*3,C
*13-11,R(C),S(C)):: CALL HCHAR(20,C*27-24,104):: GOTO 220
300 CALL COINC(#A+2,#C+2,6,X):: IF X THEN CALL DELSPRITE(#A+2,#C+2):: P(1),P(2)=
0 :: GOTO 220
310 CALL COINC(#A,#C,8,X):: IF X=0 THEN 220 :: CALL DELSPRITE(#A,#C):: R(1),R(2)
=177
320 FOR X=1 TO 2 :: G(X)=G(X)+1 :: IF G(X)=5 THEN 360 :: CALL HCHAR(23,H(X,G(X))
/8+ 875,128):: S(X)=H(X,G(X))*8-7
330 CALL SPRITE(#X,123-X*2,X*13-11,177,S(X)):: NEXT X :: GOTO 220
340 CALL DELSPRITE(#A):: CALL SOUND(-700,-5,4):: R(A)=177 :: W(A)=INT(80*RND)+10
 :: CALL HCHAR(20,A*27-24,104):: G(A)=G(A)+1 :: IF G(A)=6 THEN 370
350 CALL HCHAR(23,H(A,G(A)),128):: S(A)=H(A,G(A))*8-7 :: CALL SPRITE(#A,123-3*A,
A*13-11,R(A),S(A)):: GOTO 220
360 IF X=1 THEN G(2)=G(2)+1
370 FOR A=1 TO 600 :: NEXT A :: CALL DELSPRITE(ALL):: CALL SCREEN(5):: DISPLAY A
T(5,1)ERASE ALL:RPT$("=",36)&" GAME  OVER "&RPT$("=",36)
380 IF G(1)=G(2)THEN DISPLAY AT(12,1):"SIE SIND BEIDE GENAUSO MIES": :"ALS UBOO
T KOMMANDANT": :"ES GIBT KEINEN SIEGER" :: GOTO 400
390 DISPLAY AT(12,1):USING "SIEGER : SPIELER #":-(G(1)<6)-2*(G(2)<6)
400 DISPLAY AT(23,1):"NOCH EIN SPIEL ?  <J>  <N>"
410 CALL KEY(3,K,A):: IF K=74 THEN RUN 550 ELSE IF K<>78 THEN 410
420 STOP
430 CALL COINC(#A,153,A*168-127,6,X):: IF X=0 THEN RETURN ELSE CALL GCHAR(20,A*2
5-21,X):: IF X<>45-A THEN RETURN
440 CALL GCHAR(20,A*25-21,X):: IF X<>45-A THEN RETURN
450 CALL SOUND(-500,131,6,555,6,192,6):: W(A)=INT(90*RND)+1 :: CALL HCHAR(20,A*2
7-24,104):: RETURN
460 CALL CHAR(49,"3C7E6666667E3C001838381818183C003C7E660E1C3E7E003C7E660C667E3C
")
470 CALL CHAR(52,"606C6C7E7E0C0C007E7E707C0E7E3C003E7E707C66663C007E7E0E1C38707"
)
480 CALL CHAR(56,"3C66663C66663C003C66663E0E3E3C",61,"00007E7E007E7E",63,"003C7E
660E180018")
490 CALL CHAR(65,"3C7E66667E6666007C66667C66667C003C7E6660667E3C",68,"787C666666
7C78007E7E6078607E7E007E7E607860606000 3C7E606E667E3C")
500 CALL CHAR(72,"66667E7E666666003C181818183C3C000606060 6667E3C00666C7870786C66
")
510 CALL CHAR(76,"60606060607E7E0042667E7E66666600667E767E6E6666003C7E6666667E3C
")
520 CALL CHAR(80,"7C66667C60606000000000007E7E0000007C66667C786C66003E7E703C0E7E7C
")
530 CALL CHAR(84,"7E7E18181818180066666666667E3C00666666663C1800636363 6B7F3E
")
540 CALL CHAR(88,"63773E1C3E7763006666 7E3C181818007E7E0E1C387E7E")
550 CALL CHAR(104,RPT$("F",24)&RPT$("0F",16),108,"FFFFFFFF"&RPT$("F0",16)&RPT$("
F",16),112,"00000000FFFFFFFF")
560 CALL CHAR(113,"F0F0F0F"&RPT$("0",17)&"F0F0F0F00F0F0F0FFFFFFFFF00000000F0F0F
0F",128,"3C4299A1A199423C")
570 RANDOMIZE :: CALL CLEAR :: CALL COLOR(1,5,2):: CALL SCREEN(2)
580 CALL CHAR(92,"0F1621293D5E40433C130A121108070000C0202020A0A0408031499A2C408"
)
590 DISPLAY AT(3,1):CHR$(128);"  BY FRANK MOTZKAU \^":"   WDPM PRODUCTIONS ]_"
600 DISPLAY AT(9,1):"h h   jlsrthhrthhrjhhm":"h h prjnsqjmjnjmjm jm":"hph oqjmjn
jnsmjnsm jm":"koq   kou  oo  oo  kq"
610 DISPLAY AT(15,1):"shhthhrh hjm  shnthhrh hjhhminrjmkqhhhjm  h hjmkqhhh jm "
620 DISPLAY AT(17,1):"pshjnmmh hjnprhohjnsmh h jm joq oo o okouqo o oo o o kq":
: : : :"SPIELSTART = ENTER"
630 FOR A=2 TO 8 :: CALL COLOR(A,16,1):: NEXT A :: CALL COLOR(10,12,1,11,12,1,13
,16,1)
640 CALL KEY(0,K,A):: IF K<>13 THEN 640 :: CALL CLEAR :: CALL COLOR(1,5,5,2,2,5,
9,8,8,10,13,5,12,16,2,13,12,5)
650 CALL CHAR(128,RPT$("18",8),117,"00180808187EFF7E0000BEC3BE00000000007DC37D00
0000001810 10187EFF7E",43,"010107070F1FFF3F80C0E0E0F0F0FFF8")
660 R(1),R(2)=177 :: CALL VCHAR(8,17,104,16):: CALL HCHAR(1,1,96,224):: CALL HCH
AR(23,31,104,34)
670 CALL HCHAR(23,31,104,34):: CALL VCHAR(8,16,104,16)
680 FOR A=1 TO 50 :: CALL HCHAR(RND*13+10,INT(RND*30)+2,42):: NEXT A :: CALL HCH
AR(19,24,104,12)
690 CALL HCHAR(20,4,44):: CALL HCHAR(20,29,43):: CALL HCHAR(20,30,104,12):: CALL
 HCHAR(21,24,104,18):: CALL HCHAR(22,24,104,11)
700 CALL HCHAR(23,3,120,4):: CALL HCHAR(23,27,117,4):: CALL HCHAR(20,7,128,3)::
CALL HCHAR(20,24,128,3):: CALL HCHAR(23,7,128,3)
710 S(1)=49 :: CALL HCHAR(19,1,104,9):: CALL HCHAR(23,24,128,3):: P(1),P(2)=0 ::
 G(1),G(2)=1
720 FOR A=1 TO 5 :: H(1,A)=8-A :: H(2,A)=25+A :: NEXT A
730 S(2)=201 :: W(1)=INT(40*RND)+10 :: W(2)=INT(40*RND)+10 :: CALL SPRITE(#1,120
,2,177,49,#2,117,15,177,201):: GOTO 140
```

Combas

Dieses Programm läuft auf dem TI 99/4A mit Extended Basic und Diskettenlaufwerk. Es packt auf sinnvolle Weise möglichst viele Statements eines unter DSK1.WORK MERGE abgespeicherten Programms in eine Zeile und speichert das Ergebnis unter DSK1.CBAS MERGE ab. Man kann beim Programmieren ruhig ein Statement pro Zeile schreiben (der Übersicht wegen), weil "Combas" die Geschwindigkeitsoptimierung übernimmt. Danach laufen TI Basic Programme sehr viel schneller.

Zu beachten ist allerdings, daß "Combas" Unterschiede bei der Zeichendefinition von TI Basic zu Extended Basic nicht berücksichtigt! Falls das zu bearbeitende Programm die Zeilennummer 200 nicht enthält, ist sonst eine REM Zeile mit der Nummer 200 einzufügen, weil der Interpreter wahrscheinlich nach einer Kombination von RESTORE #XX OPEN #XX ins Schleudern kommt.

Diethelm Schädler

```
100 !        COMBAS
120 !  DIETHELM SCHADLER
130 !  KRANICHSTRASSE  11
140 !  6700 LUDWIGSHAFEN
160 !  BENOTIGTE GERATE:
180 !  DISK-DRIVE
190 !  EXBASIC
210 ! ZU OPTIMIERENDES PG ->
220 ! WORK (MERGEFORMAT)
240 ! OPTIMIERTES     PG ->
250 ! CBAS (MERGEFORMAT)
260 !
270 DATA 131,132,134,136,139,147,150,152,154,161,163,168
280 A$=CHR$(130):: B$=CHR$(0):: C$=CHR$(147):: A=255 :: B=163 :: C=3 :: D=0 :: E
=164 :: F=12 :: G=1 :: H=162 :: DIM ZNR$(5),CH$(12):: FOR I=G TO F :: READ CH ::
 CH$(I)=CHR$(CH):: NEXT I
290 DEF FPP(X)=INT(X/2)*2-X :: DEF FZNR$(X$)=SEG$(X$,G,2):: DEF FVGL(I)=POS(X$,C
H$(I),C):: DEF FZV(X$)=POS(X$,CHR$(201),C):: E$=CHR$(A)&CHR$(A):: OPEN #B:"DSK1.
WORK",VARIABLE 163 :: ZNR$(0)=E$
300 IF EOF(B)THEN 340
310 LINPUT #B:X$ :: IF POS(X$,C$,C)THEN CALL ZIEL(ZNRR(),FZNR$(X$),ANZ)
320 IF FL=D THEN CALL ZIEL(ZNR$(),FZNR$(X$),ANZ):: FL=G
330 ZV=FZV(X$):: IF ZV THEN X$=SEG$(X$,ZV+B,A):: CALL ZIEL(ZNR$(),FZNR$(X$),ANZ)
:: GOTO 330 ELSE 300
340 RESTORE #B :: OPEN #E:"DSK1.CBAS",VARIABLE 163
350 IF EOF(B)THEN CLOSE #B :: CLOSE #E :: END
360 LINPUT #B:X$ :: FOR I=D TO ANZ :: ZN$=FZNR$(X$):: PP=POS(ZNR$(I),ZN$,G):: IF
 PP>D AND FPP(PP)<D OR FL=D THEN 400
370 NEXT I
380 L=LEN(X$):: LX=LX+L-2 :: IF LX>H THEN 400
390 PRINT #E:A$&SEG$(X$,C,L-C);:: GOTO 420
400 IF X$=E$ THEN PRINT #E:B$ :: PRINT #E:X$ :: GOTO 350
410 PRINT #E:B$ :: LX=LEN(X$)-6 :: PRINT #E:SEG$(X$,3,LX);:: FL=G
420 FOR I=0 TO F :: IF FVGL(I)>D THEN FL=D :: GOTO 350
430 NEXT I :: GOTO 330
440 SUB ZIEL(ZNR$(),ZNR,ZNR)
450 FOR I=0 TO ZNR :: PP=POS(ZNR$(I),ZN$,1):: IF PP<>0 AND INT(PP/2)*2-PP=-1 THE
N SUBEXIT
460 NEXT I
470 IF LEN(ZNR$(ZNR))<254 THEN ZNR$(ZNR)=ZNR$(ZNR)&ZN$ ELSE ZNR=ZNR+1 :: ZNR$(ZN
R)=ZNR$(ZNR)&ZN$
480 SUBEND
```

Klarstellung!

Aus gegebenem Anlaß möchten wir hier eventuellen Mißverständnissen oder Verwechslungen vorbeugen. In der letzten Zeit sind in einigen Teilen Deutschlands die TI-Workshops wie Pilze aus dem Boden geschossen. Wir wurden vielfach angerufen und gefragt, ob wir nun Filialen eröffnet hätten. Dem ist aber nicht so. Es gibt weiterhin nur einen TI 99er Workshop Rheinland!

Wir möchten hier feststellen, daß wir mit keinem dieser Workshops mehr oder weniger zu tun haben als mit jedem anderen Club. Sollte sich der eine oder andere Workshop mit dem Hinweis auf unsere Gruppe irgendwelche Vorteile verschaffen wollen, bitten wir um eine kurze Nachricht an

TI 99er Workshop Rheinland
Dept. Allgemein & Software
c/o Mike Hemser
Karl Marx Allee 18
5000 Köln 71

Tel. 0221/70 39 79
(19.00 - 22.00 Uhr)

Ihr könnt aber auch eine Nachricht in unserer Mailbox, die selbstverständlich auf einem TI läuft, unter der Nummer 02 28/65 82 32 hinterlassen.

Es sei hier nochmals betont, wie auch schon mit den Workshops persönlich besprochen, daß wir nichts gegen die Mitglieder oder auch die Clubs als solche haben. Wir versuchen nur, unseren guten Namen gegen evtl. Mißbrauch zu schützen.

Wir werden auch weiterhin mit allen Clubs, die dies wollen, schön hoffentlich fruchtbar zusammenarbeiten. Also schreibt uns, wenn ihr als Club oder auch als Einzelperson an einer Zusammenarbeit interessiert seid.

TI Special

Tuning: 128K RAM plus Centronics-Interface

Stabiles Metallgehäuse. Versorgung über Steckernetzteil, durchgeführter Expansionsbus, 4K ROM gesockelt, 32K RAM direkt eingeblendet, ermöglicht TI Writer, Ext. Basic etc. RAMdisk, Centronics Druckeranschluß, leicht verständliches Handbuch

Best.-Nr. TI 1 **400.– DM**

Komfort: TI-Maus

Maus mit Anschlußkabel, 5 V Steckernetzteil, Software auf 5.25" Disk

Best.-Nr. TI 2 **298.– DM**

Nützlich: TMS 9900 Assembler

DAS Nachschlagewerk für TI Assembler Programmierer. Der Prozessor, seine Befehle, Beispiellistings, Speicheraufbau, Tabellen, Index

Anforderungen:
Editor/Assembler, Extended Basic

Best.-Nr. TI 3 **49.– DM**

Lehrreich: TI Assembler Kurs II

Eine Einführung in die Assemblerprogrammierung mit dem TI 99/4A.

300 Seiten inkl. Diskette

Best.-Nr. TI 6 **80.– DM**

Lang erwartet: TI-Assembler-Kurs III

Die logische Fortsetzung zum TI Assembler Kurs II. Bietet noch mehr Einblick in das Innenleben Ihres TI

Best.-Nr. TI 4 **79.80 DM**

Zeitsparend: Turbo-Assembler

Doppelt so schnell wie Editor/Assembler, weniger Platzbedarf auf Disk, schnelleres Laden von Disk

Anforderungen:
Konsole + Ext. Basic, Cassettenrecorder, 32K RAM Erweiterung

Best.-Nr. TI 5 **39.– DM**

CK SPEZIALITATEN-BESTELLSCHEIN

Anzahl	Best.-Nr.	Artikel	Preis des Artikels
	TI 1	128K Ram plus Centronics	
	TI 2	TI Maus	
	TI 3	TMS 9900 Assembler	
	TI 4	TI-Assembler-Kurs III	
	TI 5	Turbo Assembler	
	TI 6	TI Assembler Kurs II	

Name des Bestellers

Anschrift

PLZ, Ort

Datum, Unterschrift

Ich wünsche folgende Bestellung:

☐ Nachnahme (zzgl. 5.70 DM Versandkosten)

○ Vorauskasse (keine Versandkosten)

Bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe 43412-750 überweisen.

Coupon ausschneiden, auf Postkarte kleben und einsenden an

Verlag Ritter Eberle/CK-Software, Postfach 1640, 7518 Bretten

KLEINANZEIGEN

Spectrum

● ZX Drucker Metall-Papier ●
Ab 5 Rollen 4,– DM + Versandkosten 5,90 DM Nachnahme bzw. 3,50 DM bei Vorkasse oder Verrechnungsscheck. EDT-Müller, Flügelkamp 13, 3000 Hannover 21

ZX Spectrum Datenbank für Schachpartien. M Code. Es spielt und speichert über 250 Partien. Info 80 Pf bei C. Spärk Hofgasse 23a 6117 Schaafheim-Mosbach

●●●● Achtung ●●●●
Programmsammlung löst Ihre Probleme bei einer unübersichtlichen Softwaresammlung (mit 64 Zeichen pro Zeile und Quellenverwaltung). Info gegen Rückporto oder Programm direkt gegen 20,– DM (Schein, Scheck). Jürgsoft Postfach 10 06 10 4040 Neuss 1

Suche für Spectrum zuverlässige Softwaretauschpartner (Cass.). Zuschriften und Listen an: Claudia, Postfach 1233 4053 Jüchen 2. Antwort 100%ig

Sinclair-Freaks!
Ein User-Club informiert
Seit 2 Jahren besteht der Spectrum-User-Club Wuppertal. Unsere monatlich erscheinende Clubzeitschrift enthält neben Erfahrungsberichten über Hard- und Software auch Tips, Tricks, die Mitgliederbörse und vieles mehr. Außerdem kann jedes Mitglied eine monatlich erscheinende Kassette mit den besten von Club-Mitgliedern entwickelten Programmen beziehen.
Wer mehr über diese und weitere Leistungen des Clubs wissen will, kann gegen Rückporto mehr Informationen anfordern unter der Clubanschrift: Rolf Knorre, Postfach 20 01 02, 5600 Wuppertal 2 ● Angebote von Händlern für Club-Rabatte sind erwünscht

Spectrum ●● Schnelle SAVE und LOAD bis zu 4000 Baud, nur 30,– DM; 2-Pass-Ass./Disass. MDV/Disk-komp., 100 Zeilen/sec. Trace alle Inoff. Befehle nur 40,– DM. Bei M. Strahm Rölscher Straße 156/1513 5100 Aachen

Super-Angebot: Currah µSpeech Interface 1, Microdrive, 24 Cartridges, für nur 300,– DM (zusammen) auch einzeln zu verkaufen! ☎ 0 22 21 / 32 06 (ab 18 Uhr)

● ZX Spectrum Video Interface ●
Anpassungsfähig (regelbar), inverse Darstellung über Hardware. Reverfläche. Nur an den Userport stecken. Preis 70,– DM inkl. Porto + Verpackung ☎ 0 71 45 / 55 80 ab 18 Uhr

Ich programmiere alle EPROM-Typen für Spectrum, QL, ATARI ST und alle anderen Computer. Anfragen unter ☎ 09 11 / 44 60 00 ab 18 Uhr, D. Appel

Achtung Spectrum User!
Tausche Software für Spectrum 16 – 48 K. Suche User, die mit Discovery 1 DOK Erfahrung haben. Heike Kemphausen, Postfach 1133, 4053 Jüchen 2 ☎ 0 21 64 / 76 10 nach 18 Uhr

Suche für ZX 81 seriellen Interface RS 232 von Memotech ☎ 0 90 52 / 60 23

●●● Achtung Spectrum-Freaks ●●●
ZX Spectrum 48 K, CUB Farbmonitor 452 (TTL und Spec.-Eingang), Spec. Interface 1, Sprint, Dig.-Recorder, ZX Lprint II mit Parallel- und Seriellkabel, Orig. Software u. a. Beta Basic, Masterfile mit MF-Print, F. ght, Skutils, günstigst (auch einzeln) abzugeben. P. Enders, ☎ 0 91 31 / 6 39 27, 1 61 15 oder 91 27 51

● ZX ●●● Na endlich ●●● ZX ●
Ich übertrage Euch jedes Programm auf Microdrive. Nur 2,– DM pro Programm + Versandkosten. Schickt Eure Programme und die leeren Cartridges an Adresse: Uwe [illegible], Mozartstr. 21, 5800 Hagen 5 ☎ 0 23 34 / 5 80 47 ● Ich schicke Euch die Cassetten und die vollen Cartridges per Nachnahme wieder zurück!

●● Suche für ZX Spectrum und QL **●●** Selbstgeschriebene Programme zu kaufen (Anwendersoftware und Spiele). Schickt mir mal eine Liste (wenn's geht) mit Beschreibung an: Dataland, Lärchenweg 5, 4290 Wese

SPECTRUM-PROGRAMMSAMMLUNG (I)
5 Anwendungen, 5 Spielprogramme. **TOOLBYTE** 100% MC 5 KBytes, 20 anfängerfreundliche, starke und effektive Erweiterungsbefehle. **LINE-DELETE** 50 Bytes MC. Löscht Zeilen einzeln u. bereichsweise. **COMPACTOR 122 Bytes MC. Verkürzt alle! Basic Progr. ca. 60%. HELP** 200 Bytes MC. Verlängert laufende Progr. mit Anzeige d. Zeile. **DRAW Grafik 3,5 KBytes MC 19 Befehle**, Circle, über 1000 versch. Schrifttypen, 3 Gummibänder. **TECHNO** 5 KBytes. Erstellt techn. Zeichn. in 3 Ansichten + Perspektive – mit Demo. **U-BOOT-SIMULATION** 40 KBytes MC 17 Befehle u. Kontr. + Kontrollraum, Navigationsraum u. Periskop. **CORNAZOIDS** 6 KBytes MC. Sehr schönes Arcade. **SNAKE-ALIVE** 6 KBytes MC 40 Ebenen. **QUAZIMODO** 3,5 KBytes MC. Aufwärts 4 Screens. **BLOCKMAN** 8 KBytes. Kraft und Spannung. Alle 10 Programme in kommerzieller Qualität aus Your Computer 84/85 mit guter deutscher Dokumentation **DM 25,–** an Josef Bauer, Jakobstr. 11, 6650 Homburg. Jede weitere [illegible] Dok. [illegible] 4,– DM

●●● Hurra ●●●
Drosseln Sie den fast freien Spectrum per Software. Anwendungsgebiet: Debugging, Bremsen von Spielen etc. Info gegen Rückporto oder Bestellung (Cass. und ausf. Anleitung) – 25,– DM bei Jan Wellerstr. 35, 5630 Remscheid

Endlich ist es da!
Die 64-Zeichen-Karte für Ihren Spectrum. Dieser Hardwarezusatz bringt 64 × 32 Zeichen auf den Schirm. Kompatibel zu nahezu allen Anwendungsprogrammen (Tasword, Devpac etc.) Gratisinfo bei: Jan Wellerstr. 35, 5630 Remscheid

DL ART – das 900 und Zeichenprogramm! Das neue Grafikprogramm für den QL. Komplett mit Font-Editor, Zoom 2, Bild im Speicher, Hardcopy, Fill, Shapes, komfortable Menüführung, kopieren von Bildteilen, sehr schnell! Handbuch und Cartridge für nur 98 DM. Auskunft ☎ 0 40 / 7 01 92 18 oder 0 40 / 7 01 74 36

●●● Verkaufe komplett **●●●**
Spectrum 48 K + Kempston-Maus + Kempston Centronics-F + Plotter Schreibmaschine (4 Farben) + Joystick + IF + Software Art Studio, Tasword 2, Planets, Laser Basic, Winter Games, 3D-Uhren + Datenrecorder, alles 3 Monate alt, Orig. Verp. NP 1400 DM für 1200,– DM ☎ 0 61 21 / 6 82 74

● Einkommen-/Lohnsteuer 1986 ●
Schon jetzt die Steuer 1986 berechnen und beeinflussen, die Jahresauswertung möglich. Update-Service 1987 ff. Spectrum Cassette 50,– DM, Microdrive 55,– DM. Info 80 Pf. Auch C 64/Apple II. Dipl. Finanzw. R. U. Dufs, Bachstr. 70, 4216 Niederkassel 2 ☎ 0 22 08 / 4 61 56

ZX-Spectrum 80 KB, ISO ROM + Text, Recorder, DP 50 E Digital Tracer, viel Software und Literatur. Preis VHB ☎ 0 75 51 / 6 61 56

Suche defekten Sinclair QL oder Spectrum. ☎ 0 63 57 / 70 37 ab 18 Uhr

Zeichenbank mit echt Spectrum. 200 Byte machen aus Screens den Font und 150 Byte zeigen ihn. Preis 10,– DM Peter Pfaffmann, Gutenbergstraße 16, 7340 Geislingen

ZX Sinclair Spectrum Videointerface für alle Monochr. Monitoren. Hardwaremäßige inverse Darstellung (Regelbar). Einfach Taste nur an den User-Port anstecken. Preis 70,– DM inkl. P+V ☎ 0 71 80 / 58 80 ab 18 Uhr

●●● Spectrum 48 **●●●**
2-Pass-Ass. und Disass. 40,– DM. 2-Pass-Macroass., Disass. und Reass. 50,– DM. Mdv/Disk-komp. 100 Zeilen/sec. Turbo-Save/Load bis 6000 Baud 30,– DM. M. Strahm, Rütscherstr. 156/1513, 5100 Aachen

Lotto-Statistik 6 aus 49 Spectrum 48 K. Alle Ziehungen seit 1955. Auswertung in 10 sec! Mit Gewähr! Es sind 3 Bank- und Vierling viele Auswahlzahlen können untersucht werden. Cass. 45,– DM Scheck oder + NN G. Ritz, Wiesenstr. 10, 5650 Solingen. Auch für CPC 464, 664 und 6128

Drucker GP 50 S zu verkaufen. Preis 200,– DM VHB. C. Rewertherm, Heckamp 7, 4832 Ver[illegible] ☎ 0 52 48 / [illegible] 50

Verkaufe Spectrum mit Saga 1 Tastatur, Discovery 1, Drucker CP 80, Multiface one, Disketten, viel Software, zahlreiche Bücher (Prog. + Hardware), ZX 81, Pascal auf dem Spectrum, Preis komplett 1000,– DM. Rüdiger Velten, [illegible] 11, 7133 Maulbronn 2, ☎ 0 70 43 / 6 99

Verkaufe ZX Spectrum + Microdrive Interface 1 und viele Programme für 900,– DM ☎ 0 64 42 / 4 80

●●● An alle Spectrum-User **●●●**
● Werfen Sie Ihr ROM weg ●
Von mir bekommen Sie ein **besseres**. Mit Save eigener Zeichensatz und viele neue Routinen für nur [illegible] DM. INFO anfordern: S. Hoffmann, Am Rain 6, 3305 [illegible] 1

● Verkaufe ● Verkaufe ● Verkaufe ●
Kempston E Centronics IF VHB 120,– DM, Sinclair Interface 2 VHB 30,– DM. Beide Interfaces sind nicht beschädigt, mit Anleitungen. Suche Kontakt zu anderen Sinclair-Usern. Thomas Höh, Mittelweg 2, 6365 Bad Rosbach ☎ 0 60 03 / 81 74

QL – Verkaufe meine Originale für den QL. Wanderer (Pyramide) 45,– DM und Super Astrologer de Luxe (V 1.4) nur 60,– DM ☎ 0 23 81 / 2 46 20 (abends)

● ZX Spectrum Maschinencode ● neu nur 30,– DM ● Alles über Sinclair Comp. ● neu nur 30,– DM, deutsche Original Handbücher nur 60,– DM ☎ 0 51 94 / 27 52

Schüler sucht günstig Microdrive! Angebote an Michael Dehe, Dielingen 454, 4995 Stemwede ☎ 0 54 74 / 3 18. Eventuell **Tausch gegen 17 Original-Programme.**

17 Original-Programme (Spectrum 48 K) VHB 90,– DM. ☎ 0 54 74 / 3 18 Meldet Euch!

ZX Spectrum 16 K / 48 K / + / 80 K / 128 K Software ab 50 Pf pro Programm. Info gratis: Wolf Lutz, Mittelweg, Postfach 49, 3256 Aerzen 1

Verkaufe Super Software z. B. Night Shade, Exp., Fist, Spy v. Spy und viel mehr. Bitte schnell ☎ 0 89 51 / 1 80 60 anrufen!

●●● Verkaufe ●●●
Sinclair Spectrum 48 K, Datronics Interface, HiSoft Pascal, C, Assembler, Toolkit, Prolog, Originale mit Handbüchern für 300,– DM ☎ 02 09 / 59 37 69 10 bis 14 Uhr

● Wegen des Erfolgs ●
Spectrum-Super-Info-Power
900 K B, brandneu und nur 10,– DM ● Teile Video! ● nur noch 4,– DM ● Top-Info **1,– DM Bei Lore Spectrum-Club Hohlstraße 11 6791 Steinbach/Glan**

Suche Int. 1, Cartridges mit 128-K-Programme. **Verkaufe** Spectrum, def. und Micro Hefte VHB. **Suche** auch Kontakt zu User-Clubs in der Gegend! Harald Hoffmann, Brunnenweg 38, 6700 Ludwigshafen

● Hallo Spectrum-User! ●
Ich löse meine Softwaresammlung auf, und dabei gibt es garantiert eine Menge Schnäppchen: Aktuelle Topgames zu Superpreisen. Eine Liste gibt's gegen 80 Pf Marke. Jürgsoft, Postfach 10 06 10 4040 Neuss 1

●●●● Verkaufe ●●●●
Microdrive Interface 1, 16 Cartridges + Trans Express Kopierprogr. + Listing, Games usw. zus. 250,– DM. Stephan [illegible], 6000 Frankfurt Main ☎ 0 69 / 44 74 79 ab 20 Uhr

Goodbye Sinclair!
Den Rest aus meiner Systemaufgabe jetzt zum Sonderpreis (alles VHB): Microdrive Cartridges à 3,90 DM, Trans Express (auf Cartr.) 15,– DM, Druckerinterface Kempston E 95,– DM, Literatur ab 5,– DM, Software (org. Cass.) ab 1,– DM. Es lebe Atari! auch ZX 81! [illegible] DM Gratisinfo/Bestellung/Angebot: H. Pössner, Dachauer Straße 483, 8000 München 50 ☎ 0 89 / 1 50 71 60 ebenda Tausch gegen CPC Software erwünscht!

ZX-SPECTRUM 48-K zu verkaufen mit Tronic-Tastatur (12er Block) + über 150 Programme von Tennis bis Decathlon + eingebautes Joystickinterface + S/W-Fernseher + Seikosha GP 100 A Matrixdrucker + Druckerinterface ZX LPrint II + Bücher + Zubehör für zusammen nur 850,– DM. Lenkhard Fuchs, Staufeneckstraße 16, 7340 Geislingen/St. ☎ 07331/67651

Verkauf von ZX-81 / 16 K-Software Info: H. Lange, Hochkreuz 25, 7777 Salem-Beur

●●● Hallo Spectrum-User ●●● Zu verkaufen: 1 Microdrive (60,– DM) + Interface 1 (70,– DM) + 17 Cartridges (je 6,– DM). Alles zusammen 210,– DM. Suche Kontakt zu Opus-Usern und zu Programmierern. Ruh, Jürgen, Postfach 100610, 4040 Neuss 1

Suche Programme für Spectrum + für Wetterstation ELV und ELV 961. 8600 Dieter Kruse, Schillerstraße 15, 8600 Bramsche 1

Verkaufe ZX Spectrum 48 K, ZX Printer (defekt), Seikosha GP 500 A, diverse Literatur und Software zu günstigen Preisen. ☎ 02301/3458 nach 18 Uhr

●● Tausche Spectrum-Spiele ●● (Cass.) gegen Atari (Disk/Cass.) oder Spectrum-Programme (Cass.). Liste an Andreas Rengsmacht, Ahornstraße 6, 5760 Arnsberg 1

●● Hallo Spectrum-Profis ●● Suche Software aller Art. Suche auch guten Drucker! Schreibt an B. Jungschöffel, Herrenweg 10, 2965 Ihlow-Ostersander ☎ 04955/5 74

ZX-Spectrum + Profitastatur + Kempst. Joy-IF + viel Literatur + Software. VHB 480,– DM ☎ 08848/696

● Sinclair QL ● Tausche bzw. verkaufe Originalprogr. Machine Code Editor Assembler (McGraw Hill) 70,– DM, Flight Simulator (Microdeal) 40,– DM. Gesucht: Psion Chess, QL Art u. a. Lerch, Schloppbühel 38, 5480 Linz

Sinclair Spectrum + Opus 180 K + Tastatur + Disketten + Software + Literatur komplett gegen Gebot ☎ 07249/8875 ab 19 Uhr

Verkaufe Spectrum 48 K + Microdrive + Interface 1 + HiSoft Pascal und Assembler + div. Spiele und Bücher. Alles im Zustand und orig. verpackt für 400,– DM ☎ 05401/1584 nur bis 20 Uhr So.

● Verkaufe Beta-Disk-Controller ● Ausführung 4.12 für 880,– DM. Siegfried Hauger, Hauptstr. 88, 7734 Brigachtal

Verkaufe Interface 1 - 3 M'Drives, 50 Cartr., ZXL Print 3, Monitorinterface. ☎ 08142/2084

Spectrum 48 K + gute Tastatur + Interface 1 + 2 Laufwerke, alles mit Netzteil im Tastaturgehäuse. Ferner Drucker GP 50 S, 200 Cassetten und 12 Cartridges mit vielen guten Programmen, viel Literatur. Komplett für 600,– DM VHB. ☎ 0281/35405

●● Spectrum 80 k, Kompaktbau **●●** inkl. Kass.-Laufwerk, Opus-Controller, Floppy 5 2¼ Zoll 1 MB, Sage 1 Tast., Sound- + Joyst. CK Interface, Reset, Centronics, Disketten, Software-Literatur + Handbücher kompl. 800,– DM. ☎ 02 06/ 4452 74 bis 18 Uhr

●●●● Top! ●●●● **Biete Software, Bücher und Magazine u. a. m. zu Spottpreisen! Thomas Schröer, Hohlstraße 11, 6791 Steinbach. Info 1,– DM! Centr. Drucker-Kabel (Opus) 25,– DM!**

Kempston Centron. E + LPrint III, je nur 85,– DM! ☎ 05963/7490 ab 18.30 Uhr!

Verkaufe Spectrum 48 K, Fuller FDS, mit Beta Disk Contr. 4.7 DUS u. Drucker Int. (1 Kempst. 1 ZX LPrint), Lichtgriffel, Tastaturfolien, Ersatz-Ula-Spule, Tasword, Masterfile-Tasprint, Tasmerge, RTTY Soft- und Hardware-Decoder 10 auch auf Diskette. Preise VHB, auch Einzelverkauf ☎ 04195/517

●●● Bücher für ZX-Spectrum ●●● 50% unter Neupreis: ZX Spectrum Tips + Tricks von Data Becker, weitere Kniffe und Programme mit dem ZX-Spectrum, Maschinencode von Bellhouse, Basic-Handbuch und Pascal auf dem ZX Spectrum ☎ 06253/4710 ab 16 Uhr

Atari

●●●● ATARI 600 XL / 800 XL ●●●● 256 KB RAM-Speichererweiterung, XE kompatibel inkl. Hilfsroutine Software:
a) RAMDISK (alle Atari-Formate)
b) Kopierprogramm (1 Diskwechsel)
c) Adreßverwaltung (2000 Ad.) einem Preis 278,– DM. Info gegen Rückporto bei Volker Rasche, Gallistr. 22, 4970 Bad Oeynhausen 1 ☎ 05731/22506

ATARI 260 ST 1 Megabyte Aufrüstung, steckbare Platine ohne Löterbeiten, Eprom-steckkarte und User Port Karte. Für 520/260 ST. Das Rev. B und es laufen alle Programme. **Happy 1050** und Highchip ☎ 02 02/4022 14

Verkaufe Original-Diskette Atmas II, Makroassembler mit Anleitung 26,– DM, Originalcassette Forbidden Forest für nur 10,– DM ☎ 0221/57281

Suche Atari-Software, nur D. Z. B. Gunship, Gulat und gute Anwender-software. Tausch möglich z. B. Silent Service, Print Shop, Wasser! usw. Liste an R. Lobert, 4700 Hamm 3, W. v. Braun-Straße 8

Seikosha GP 100 A für Atari 800/800 XL, Direktanschluß, 6 Monate alt, wenig benutzt – 150,– DM ☎ 030/4631675

●● Achtung Software-Besitzer ●● Suche dringend – Simulationsprogr. einer CNC-Steuerung (auf 800 XL). Zahle gut. Angebote bitte an J. Frekasch, Sudetenstraße 10, 8765 Erlenbach ☎ 09372/6478 ab 18 Uhr

Verkaufe Abfrage- und Übersetzungsprogramm für Atari 800 XL. Es sind über 4000 englische und deutsche Wörter gespeichert. Cassette für 20,– DM bei J. Stein, Hinderstr. 4, 8263 Burghausen

● Atari XL Software auf Cassette ● ab 50 Pf. keine Raubkopien. Liste gegen 80 Pf Rückporto bei Michael Wagner, Altenburger Str. 44, 6430 Alsfeld

ATARI XL/XE Tausche Software! Habe z. B. Int. Karate, Last V8. ☎ 09372/9049 ab 18 Uhr

Verkaufe Rändchen F+ mit Interface für Atari (Centronics) 100% Epson-Kompatibel, Matrix NP 1100,– DM, ½ Jahr alt, VHB 800,– DM ☎ 02625/4612

Wir suchen einen Redaktionsvolontär!

Die CK-Computer Kontakt erscheint jetzt schon im 3. Jahrgang und hat bei den Lesern der Geräte Sinclair, Atari und TI 99/4A einen festen Platz erreicht und befindet sich weiter im Aufwind. Nicht zuletzt verdanken wir diesen Erfolg unseren hervorragenden Mitarbeitern. Für den weiteren Ausbau der CK-Computer Kontakt, des Schneider-Magazins sowie unseres neuen Atari-Magazins suchen wir jetzt einen Redaktionsvolontär, der mit uns an den bestehenden Objekten arbeiten und auch neue Ideen angehen will.

Wir stellen uns einen Computerfreak mit Begabung zum Schreiben vor. Er sollte gute bis sehr gute Kenntnisse in Basic, Assembler und MS-DOS vorweisen können oder sich zumindest zutrauen, fehlende Bereiche schnell zu erlernen. Besonders wichtig ist uns aber, daß er weiß, was er will und wir uns auf ihn bei allen Arbeiten verlassen können. Seine Computererfahrungen sollte er bevorzugt auf Atari 800XL/130XE oder Atari ST gesammelt haben.

Das ideale Alter unseres neuen Redaktionsvolontärs wäre ca. 25 Jahre, er kann aber auch älter oder jünger sein. Was zählt, ist die Qualifikation. Beginnen könnte unser neuer Mann sofort bei uns, in unseren Büroräumen ist genug Platz. Die Stelle wird nach Tarif vergütet.

Wenn Sie diese Aufgabenstellung reizt und sich vorstellen können, Ihr Hobby zum Beruf zu machen, so wenden Sie sich schriftlich an uns.

Verlag Rätz-Eberle, Postfach 1640, 7518 Bretten

Flensburger Computer Versand

M.-L. Sander · Am Sout 4 · 2390 Flensburg · 0461/32533

5¼"-Disketten 2D 10er **10.–**
3½"-Disketten 1D 10er **39.–**
3½"-Disketten 2D 10er **45.–**
3" Maxell CF2 10er **85.–**

Ab 300,– DM bequeme Teilzahlungsmöglichkeit.

CDI Hitrans mit NT Muller **222.–**
DFÜ-Interface mit SW f. Spectrum **100.–**
Spectrum Grafiktablett (das Grafitel) **400.–**
Mirage Microdriver **100.–**
Philips MSX 2 VG8235 **1450.–**

●●●● ATARI ●●●● Ich biete für 10,– DM ein Malprogramm an. Die gemalten Bilder, die man natürlich abspeichern kann, lassen sich für selbstprogrammierte Spiele verwenden. Wie? Das steht in der Beschreibung, die mitgeschickt wird. Schicke 10,– DM und Deine Adresse an: Werner Dersching, Neubachstr. 42, 8520 Wasserburg

● Atari 130 XE ● Suche Module, Software, möglichst auf Diskette, vor allem gute Spiele. Suche auch Farbmonitor. Informationen bitte an T. Karasayir, Osterstraße 47, 4700 Hamm 1

Suche Kontakt zu 800 XL-Usern! Habe selbst einen 130 XE + Floppy. Meldet Euch bei Jörg Spitzner, Königstr. 32, 6750 Kaiserslautern

Verkaufe Computerzeitungen diverse ATARI-Listings. Listen gegen 2,50 DM in Briefmarken von H. W. Herrmann, Mehrweg 21, 5628 Heiligenhaus

Atari 800 XL. Suche Programme aller Art (Disk). R. Müller, Auferweg 16, 5600 Gevelsmarkt

Atari XL/XE Software! Verkaufe wegen Systemwechsel meine komp. Programme zu 40–70% des NP. Beispiel: Print Shop, Typesetter, Ultima 3, Arcade Mach. Nur Originale, keine Raubkopien! Liste anfordern bei: M. J. Meyer, Kreuzmattweg 15, CH-4144 Arlesheim/Schweiz. **Atari!**

Verkaufe Atari 800 XL + 1010, Progr., div. Bücher, Steck. VHB 500,– DM. ☎ 07129/50189

extrem stabile
JOYSTICK'S
(Acht-Wege-Fahrhebel)

- [illegible] bewährt im härtesten **SPIELHALLEN-EINSATZ**
- passend für: C 64, C 128 / Atari 400, 800, 130 / Schneider CPC / SVI 318, 328 u.v.a. auch in Ausführung für TI-99/4A
- von [illegible] MÜNZ-AUTOMATEN-HERSTELLER [illegible] gefertigt
- alle Schaltkontakte [illegible] über Mikroschalter
- **1 Jahr Garantie - DM 46,-**

Lieferung kurzfristig per NN plus DM 5,50 Porto und Versandkosten, oder gegen Vorkasse DM 46,– (porto- und verpackungsfrei) Postgiro-Konto Köln 369623-5[illegible] oder durch Verrechnungsscheck

AUTOMATEN - SERVICE ALAIN KRAWIETZ
BAUERNBANKSTR. 27 TEL. 0221/37 23 49 5000 KÖLN [illegible]

Habe Lösungen zu: The Dark Crystal, Atlantis, Cranmore House, The Mask Of The Sun (Dt.+Engl.), Null Grad Nord, Schloß des Grauens, Abenteuer im Weltraum, The Serpent's Star, The Dallas Quest, Nibelungen. Je 10,–DM Schein beilegen! Zu richten an: D. Klich, Weberweg 2, 8590 Marktredwitz

Verkaufe ATARI-800 XL-Software. Liste bei Hat Fistoks, Palmstraße 47a, 4250 Bottrop. 80 Pf. Rückporto!

800 XL ●● 8 + 4 Programme ●● wegen Systemwechsel zu Schleuderpreisen zu verkaufen. Nur Disk. Gratisliste anfordern! Karl-Heinz Hartmann, Casa Ginestra, CH-6518 Gorduno, Schweiz

●●●● Staubkiller ●●●●
Abdeckhauben für Atari 800/800 XL/1050, C64 15●1, Klemklit 14,95 DM, Kunstleder 19,95 DM. VK oder NN. Bei M. Kühn, Ummerstr. 1F, 4100 Duisburg 17, ☎ 0 21 36 / 51 9 77 nach 17 Uhr. Bei Vorauszahlung keine Gebühr

Super Lotto-Programm geschrieben auf ATARI 520 ST+. Mit VEW Systemen, eigene Ziehungen. Das nie-Tadellen [illegible] gesagt. [illegible] Programm mit vielen Möglichkeiten. 50,– DM oder 40,– sFr. Einsenden an D. Steiger, Wilf[illegible]straße 164, CH-4056 Basel

●●●● ATARI ST ●●●●
Software für Erwachsene (Porno), Bodythrous, Reversi, Solitaire u.v.m. Info anfordern bei: Soft, Postfach 1619, 4810 Gütersloh

Verkaufe Atari-4 Farb-Plotter 1020 Direktanschluß, VHB 220,– DM. ☎ 040/496 28 69

● ATARI ●●● RABE ●●● ATARI ●
Ein Programmer, Textunterstützungs-, Fehlersuch- und Programmanalysesystem ermöglicht Einblicke in ungeschützte Software. Per Tastendruck jederzeit aufrufbar. Preis 75,– DM EPROM + Disk. Info bei A. Kern, Glanzstraße 5, 8051 Hachenreihing, ☎ 08071/1225

● Lohn- und Einkommensteuer ●
Super Jahresausgleich mit Anzeige, jährl. Aktualisierung ohne Neukauf. Lieferbar [illegible] Diskette 70,– DM für Atari. Info gegen Rückporto von H. [illegible], Niederfeldner Straße 44, 8072 Manching, ☎ 0 84 59 / 16 89

Atari 400/800/XL/XE
Verkaufe Spiele + Anwendungen, z.B. von SEA, Synapse, Lucasfilm auf Disk., Cass. und Modul. Preise von 5,– bis 30,– DM. Liste anfordern: ☎ 0 23 04 / 4 15 10. Verkaufe außerdem Atari 1029 Drucker mit Hardcopy-Programm (Top Zustand), VHB 360,– DM

Suche zuverlässigen Tauschpartner für Atari XL (nur Disk). ☎ 0 72 52 / 8 08 52

Verkaufe Atari 800 XL + Recorder 1010 + Floppy 1050 + 4 Module + 4 Disketten + 8 Cassetten für nur 700,– DM. ☎ 02 31 / 49 11 32

Verkaufe Atari 800 XL + Datas (+ Spiele, Progr.) sehr billig. Suche C64 + 1541 günstig. ☎ 0 28 41 / 8 08 31. **Übernehme ein! Porto!**

●●● ATARI 800 zu verkaufen ●●●
Mit Basic, sep. [illegible] Handbuchern und weiteren Büchern über den Atari. VHB 2[illegible]0,– DM. ☎ 040/82 90 82

We kaufe Atari-Software auf C+D. Info gegen 80 Pf. Rückporto von Markus Kray, Sebastianusstr. 6, 5014 Fu[illegible]heim

Suche Atari Floppy 1050 bis 300,– DM. Außerdem Software und der Hardware gesucht. ☎ 0 40 21 / 6 60 67 **nach 19 Uhr**

ATARI 1050: Suche Master DOS 2.0 und 2.5 Disk für 15,– DM pro DOS + Porto. U. Kostors, Pappelweg 22, 5205 Sankt Augustin 1, ☎ 0 22 41 / 33 63 50

●●● Atari 800 XL ●●●
Verkaufe gute MS-Spiele für 5,– DM/Cas., auch Tausch). Gewinner Decathlon. Hartmann, Günther Steinle, Beethovenstraße 1, 8881 Aufhausen. Antworte sofort!

ATARI 800 XL + 64 K [illegible] mit 3 Modulen (Atari + Command. Tennis, Pole Position) und Recorder XC 11 zu verkaufen. Preis 250,– DM. ☎ 0 40 52 / 12 80, Ferien Notruf Kont 2, 2350 Lauenburg

● **Dringend** ● Suche für Atari: Flyer Fox, Conrad C, Forth u.a. m. ☎ 0 23 73 / 7 20 23, Michael Fleri, Hasenweg 9, 5758 Fröndenberg (nach 19 Uhr)

Atari Verk. Orig.-Spiele auf Disk. Pelst, Cavelord, Bristles, je 20,– DM. Conan 35,– DM, Arcade Machine 35,– DM. Christian Bergmeister, Kriegshaberstraße 28, 8900 Augsburg 1 ☎ 08 21 / 40 90 95

Hallo XL/XE Besitzer! Ich verkaufe Teile meiner Programmsammlung auf Cassette und Diskette. Info: Ludger Karking, Reiseker 40, 4405 Dülmen. ☎ 0 25 94 / 74 00 ● **Billig!**

● **Atari XL/XE** ● **Suche Summergames 2 + Wintergames** ☎ 0 93 ●6848 **Bernd verlangen!**

● Für Atari 800 Hyperbasic (C) 10,– DM ● Atari-Grafikbuch 15,– DM ● Spielturm 15,– DM ● Alles VHB. Info [illegible] Rauch, Moreco Straße 17, 7562 Gernsbach

●●●● Atari XL/XE ●●●●
Public Domain Software ausgefeilt! Disketten voll. Preis nur 50,– DM. Anleitungen in Deutsch! Bitte DOS Version angeben. Bestellung an: Lutz Machin, Postfach 49 [illegible] 2[illegible] Aachen. Das Programm wird in einem Ordner geliefert + 1 leere Diskette extra!

●● **Hallo Atari XL/XE-Freaks!** ●●
Mehr neue Spiele, Superdisketts mit Bestseller mit ca. 40 Spielen! 20,– DM. Schreiben an J. Schwarzer, Th. Heuss-Straße 37, 8660 Münchberg! 100 Spiele 10,– DM!

●●● **Software für Atari XL** ●●●
Neueste Software aus allen Bereichen (Anwendungs-, Geschäftsprog. und jede Menge Spiele). 2 ● D [illegible] mit 15 Top-Titeln 20,– DM. ☎ 02 41 / 5 7 7 16

●●● Suche Software ●●●
Ich suche Software für meinen 47[illegible] 800 XL. Bitte nur auf Cassette. Preis nach Absprache. F. [illegible] k Rudolf, [illegible]weg 50, 7000 Stuttgart 31

800 XL Atari! Suche für meinen Atari [illegible] Musik und Software. Wenn [illegible] geht, nur auf Cassette. Schreibt Angebote an Jörg Albert, Bahnhofstraße [illegible] 4100 Duisburg 18

●●● ATARI ●●●
Verkaufe Drucker 1027 und Datasette, beide wenig gebraucht. Schickt Eure Angebote mit Preisvorstellungen an Peter Jürgensen, [illegible] 104, 6301 Heuchelheim ☎ 0641/615315

Suche DOS 2 und Astrologieprogramm für Atari 800 XL. Ludwig Hoyer, Katharinenstraße 15a, 2082 Uetersen ☎ 04122/42038 Anrufe nach 18 Uhr

● Top Aktuell für Atari XL/XE ● Music-Machine II ● Synthesizer Direct Play 2 Manuals versch. Instr. neue Superrhythmen! ● Umwandlung der Songs in Basic Programme ● 30 Demos ● Half-Interface ● Als Komplettpaket oder einzeln erhältlich zum absoluten Sonderpreis ● ☎ [illegible]/1758 oder 01135/3364 nur Sa. + So.!

Texas Instruments

1000 KB-Programme für TI 99 komplett für 100,- DM auch einzeln und viel Lit. M. Bamberger, Guardinistraße 22, 8000 München 70

Assembler Programme, Disk Editor, GPL Dessassembler, Fast-Copy für E.A., MM, XB + 32K + Disk. Info gegen Freiumschlag. M. Eichhorn, Ziegelheck 1, 6840 [illegible] 4

●●● TI 99/4A ●●●
Adressenverwaltung, Textverarbeitung + div. Finanzrechnung. Je 9,90 DM, komplett 24,90 DM zuzüglich 5,10 DM Porto und Cassette, Info 1,10 DM. W. Bennetz, [illegible] 35, 4760 Werl

Verkaufe gebraucht für den TI 99 Statistik Modul + RS232 Printer Interface GP 50 A mit 5 Rollen Papier. Ab 18 Uhr ☎ 09188/1205 bei Vitzthum

Suche neue [illegible] gute Freeware für den TI mit 32 K! ☎ 06 01/58 08 87

Verkaufe TI 99/4A + Ex. Basic + deutsches Handbuch + Rec. Kabel + Joystick = 150,- DM. Diskettenlaufwerk M PHP 1850 = 100,- DM. 32 KByte-Speichererweiterung, batteriegepuffert = 100,- DM. Module, Spiele auf Cassette und Listings. Info gratis von Markus Thermer, Schwalbenstr. 2, 3203 Hochheim

TI-99/4A + Zubehör, VHB 0750,- DM ☎ 02174/60694

Verkaufe TI 99/4A + Ext. Basic, TI Plus + MBI-Centr. + 32 KRAM-ext. + Sekundär, für 100 Mark TI + div. Bücher z. B. TI-Spez. 1 + 2 + Schaltpläne + TI-Magazine bis Nr. 1 mit Cass.-Software + weitere Cass.-Software + Joystick komplett. Preis VHS. Thomas Schmid, Kernerstraße 21, 7430 Metzingen

TI 99/4A + P-Box + RS 232 + 32 KRAM + 1 TI Laufwerk zu verkaufen inklusive Module Mini Memory + Mikrosoft Multiplan, TI Write + Datenverw. und Analyse + Text-, Datei- und Finanzbuchhalt. + TI Ext. Basic + Disk Manager II mit Zenith Monitor ☎ 07[illegible]/7745

●●● TI-99/4A ●●●
+ P-Box (32 KRAM) + Disk-Laufwerk und Centr. + Ex. Basic-Erk.-Modul + Speech-Synth. Assembler-Kurs + Bücher und Hefte OK + TI Revue + Disketten (auch Spiele) + Recorder mit Kabel usw. Preis VHS. Andreas Zoll, Weberstraße 12, 7140 Bensheim. ☎ 06251/63896 Kennbal defekt

● 32 K-Erweiterung für den TI 99/4A ● für 140,- DM Batteriegepuffert durchgehend [illegible] inkl. Akkus! Gehäuse dazu 15,- DM. Assembler Disassembler 20,- DM. Manual 5,- DM. C. Moritz, Friesenweg 32, 4133 Neukirchen-Vluyn ☎ 02845/2150 [illegible]

TI 99/4A Konsole und V 24-Karte für Exp. Box gegen Gebot bei Albrecht Eger, Amlestr. 15, 7400 Tübingen ☎ 07071/61558

TI 99/4A + Ext. Basic + div. Module + Exp. Box + Disk + 32K + Lit. Preis 1100,- DM ☎ 0651/773066

Commodore

Universelle Dateiverwaltung in Assembler für C 128/80 Zeich. Disk mit Demo-Anl. 105 für 30,- DM (s. NN). Frank Hölscher, Am Wall 22, 4461 Saerbeck

C 64, Floppy 1541, Speed-DOS-E, komplett 780,- DM ggf. auch einzeln zu verkaufen. ☎ 0621/16 89 24/ 19 89 28

20 C 64-Programme auf Disks für nur 25,- DM inkl. Versand. Vorabscheck an Uwe Föttner, Kamstr. 24, 7552 Durmersheim. Info nur gegen Rückumschlag

Verkaufe C 64 + Datasette + Software VHB 350,- DM. Suche Microdrive Cartridges! Meldet Euch nachmittags bei Felix, ☎ 0641/78195, außer an Wochenenden!

C-64 Spiele- und Anwenderprogramme (C/D) gesucht. Bitte an Frigal Dremel, [illegible] Böhmert Straße 4, 2800 Bremen 33

●●● Super Best ●●●
Textverarbeitungsprogramm verwaltet Adressen und Serienbriefe 100% MC! Läuft auf C 64, 20, 16 usw. 35,- DM. Börsenprogramm für Aktien und Commodities inkl. 55 Momentankurse, versch. Gleitdurchschnitte für C 64 35,- DM. Nur Preise + 3,- DM Porto. Bei O. Hesse, Lichtenbergerstraße 60, 6818 Mannheim

VC 20 Software zu KO-Preisen. Wo? Bei P. Rogovits, Nordwalder Str. 83, 4407 Emsdetten. Info gegen 1,20 DM. Tausche auch. Suche Floppy für VC 20! Biete bis 200,- DM. Muß funktionieren!

Suche jemanden in Berlin, der komplettes Hausverwaltungsprogramm für 50 Wohneinheiten für mich schreibt. Wirtschaftliche Gegenüberstellung, Mieterhöhung usw. Näheres mündlich, möglichst Commodore 128, keine Stümper. Gage VHS. Gerhard Schulze, Aachener Str. 41, 1000 Berlin 31

Sonstiges

●●● NEU ●●●
Music Power Club, Postfach 1129, 8216 Reit/Winkl bietet 1 Hitsingle + Clubinfo + Supercassette gegen 10,- DM-Schein! Nur solange Vorrat reicht!

Ich suche immer noch für den Intellivision: B 17 Bomber, Utopia und Treasure of Tarmin. Ich kaufe Tobbe Spiele. Ruft bitte an: ☎ 0851/52284 Gunther Söller, Dr. Winklhof Str. 60, 8390 Passau

Ich programmiere alle EPROM-Typen für Spectrum, QL, Atari ST und 86 andere Computer. Anfragen unter ☎ 0911/446000 ab 18 Uhr. D. Appel

Joysoft

Spectrum 48 128 K	Preis
2112 AD	36.00
3D Games	35.00
A View to a Kill	36.00
Airwolf	29.00
Arcade Hall of Fame	32.00
Arena	39.00
Art Studio	49.00
Asterix	32.00
Astro Clone	35.00
Back to the Future	32.00
Barry McGuigan's Boxing	32.00
Batman	32.00
Battle of the Planets	35.00
Biggles	36.00
Bobby Bearing	29.00
Bomb Jack	29.00
Bruce Lee	32.00
Camelot Warriors	32.00
Colossus 4.0	29.00
Combat Lynx	32.00
Commando	29.90
Critical Mass	32.00
Dandy	29.00
Deactivators	29.00
Elite	58.00
Equinox	30.00
Fairlight	30.00
Fighting Warrior	30.00
Football Manager	30.00
FP Compiler	35.00
Frank Bruno's Boxing	30.00
Frankie goes to Hollywood	35.00
Galvan	30.00
Ghosts'n Goblins	29.00
Gladiator	35.00
Goonies	30.00
Graphic Advent. Creator	60.00
Great Escape	36.00
Gyroscope	35.00
Heavy on the Magick	35.00
Herberts Dummy Run	30.00
Highway Encounter	32.00
Hyper Sports	32.00
Impossible Mission	32.00
Infiltrator	32.00
International Karate	32.00
It's a Knock Out	29.99
I of the Mask	35.00
Jack the Nipper	31.00
Juggernaut	30.00
Knight Rider	30.00
Kung Fu Master	29.00
Laser Basic	49.00
Laser Compiler	49.00
Legend of the Amazon W.	30.00
Lightforce	29.00
Lord of the Rings	55.00
Macadam Bumper	35.00
Mermaid Madness	35.00
Mikie	32.00
Mindshadow	32.00
Mission Omega	32.00
Monty on the Run	32.00
Movie	32.00
N.O.M.A.D	29.00
On the Run	32.00
Paperboy	32.00
Ping Pong	32.00
Pyracurse	35.00
Quazatron	35.00
Rebel Planet	35.00
Rescue on Fractalus	30.00
Revolution	35.00
Rock'n Wrestle	29.00
Saboteur	32.00
Sai Combat	29.00
Samantha Fox Strip Poker	35.00
Schizophrenia	29.00
Spitfire 40	29.00

Spectrum 48/128 K	Preis
Spy Hunter	35.00
Starquake	35.00
Street Hawk	29.00
Superbowl	35.00
Superchess 3.5	37.00
Swords and Sorcery	30.00
Tantalus	32.00
Tartan	29.00
Tau Ceti	30.00
Thanatos	30.00
The Hacker	30.00
The Neverending Story	30.00
The Way of the Tiger	30.00
Theatre Europe	30.00
They sold a Million I	35.00
They sold a Million II	35.00
They sold a Million III	35.00
Three Weeks in Paradise	30.00
Timetrax	30.00
Toadrunner	32.00
Tomahawk	36.00
Turbo Esprit	35.00
Twister	32.00
Wanted Gunfright	35.00
War	29.00
Winter Games	35.00
World Cup Carnival	30.00
World Series Baseball	30.00
World Series Basketball	35.00
Yie ar Kung Fu	35.00
Zoids	32.00
Zorro	30.00
"V"	35.00

Spectrum QL	Preis
Arcadia	30.00
Better Basic Expert Syst.	59.00
Black Knight	39.00
Cartridge Media Manager	60.00
Cosmos	49.00
Droidzone	30.00
Eye-Q V2.0	75.00
Mortville Manor	60.00
Nucleon	60.00
Othello	60.00
Paintre	60.00
Professional Astrologer	150.00
QL Assembler	60.00
QL Bounder	30.00
QL Cavern	30.00
QL Dictionary	49.00
QL Jabber	30.00
QL Macro Assembler	119.00
QL Meteor Storm	30.00
QL Monitor	60.00
QL Paint	60.00
QL [illegible]	[illegible]
QL [illegible]	[illegible]
QL Touch'n Go	30.00
[illegible]	[illegible]
Sky I	[illegible]
Super Astrologer	[illegible]
Super Astronomer	[illegible]
Super Backgammon	[illegible]
Super Media Manager	[illegible]
Super Monitor Disassembler	[illegible]
Super Sprite Generator	[illegible]
Supercharge Compiler	[illegible]
Superforth	[illegible]
Technic QL	[illegible]
The [illegible]	[illegible]
The Last [illegible]	[illegible]
The lost Kingdom	[illegible]
Turbo	[illegible]
Turbo Toolkit	[illegible]
[illegible]	[illegible]
West	[illegible]
Workbench	[illegible]
Worm	[illegible]

Super! „Mein Homecomputer" 11/83–12/86 komplett 42,– DM ab mit 180 Heften nur 80 Pf T. Schröer Hohlstr. 11 6791 Steinbach ☎ 0 [illegible]/7490 ab 18.30 Uhr

Drucker HRS Thermo & Normalpapier Centronicsschnittstelle VHB 200,– DM ☎ 0 [illegible]

Computer-Kontakt das Heft mit den preisgünstigen Kleinanzeigen

Gewerbliche Kleinanzeigen

Quick Layout
Leiterplattenentflechtung für den Sinclair QL. Auto- und manuelle Routen, ein- und doppelseitige Platinen. Druckertreiber für Epson-kompatible Drucker DM 79,–

Spectrum Multiport
Interface mit 3x8 Ein-/Ausgängen, 8-Kanal-Logikanalysator (bis 10 kHz), Centronics-Schnittstelle DM 12[illegible],–
Informationen oder Bestellung bei Soft- & Hardware Johann Gollmann, Rehnbachstr. 19, 8057 Eching

Quick Layout
Leiterplattenentflechtung für den Sinclair QL, Auto- und manuelle Routen, ein- und doppelseitige Platinen. Druckertreiber für Epson-kompatible Drucker DM 79,–

Spectrum Multiport
Interface mit 3x8 Ein-/Ausgängen, 8-Kanal-Logikanalysator (bis 10 kHz), Centronics-Schnittstelle DM 125,–
Informationen oder Bestellung bei Soft- & Hardware Johann Gollmann, Rehnbachstr. 19, 8057 Eching

Preisermäßigung für Tastatur Eva Neuer Prom DM 220,– + Versandkosten E. Zupancic Postfach 2553 5810 Witten ☎ 0 23 02/5 67 61

**●●●● Sinclair QL ●●●●
Hard- und Software**
Z.B. extrem schnelle QL-RAM Disk DM 59,– 512 K-Erweiterung mit durchgeführtem Bus DM 374,– DFÜ Modapter DM 129,– Druckerinterface DM 89,– Über 100 Programme ab DM 25,– Disketten Upgrades nur DM 56,– usw. Sofort Info anfordern bei Firma Uwe Fischer Postfach 10 21 21 2000 Hamburg 1 ☎ 0 40/ [illegible] 04 61

● Disketten ●
● 5¼" 48 tpi DM 0,95 2D ●
● 5¼" 135 tpi DM 3,18 1DD ●
● 3" Schneider DM 5,85 ●
● Auch andere Garantie ●
● Allg. Austro-AG Ringstr. 10 ●
● D-8057 Eching ☎ 0 81 33/61 10 ●

**●●●●● Commodore ●●●●●
C 64 / C 128 / VC 20-User!**

Lichtgriffel mit Programmen	48,–
Resetaster (C 64/VC 20)	7,–
Akustikkoppler Dataphon s21d	342,–
Dataphon s21d mit Software und Kabel für C 64/C 128	298,–
80 Zeichen-Karte für C 64	228,–
64 KByte-Erw. für VC 20	172,–

Versand gegen Scheck/Nachnahme. Weitere Artikel in unserem Gratisprospekt. Fa. Schößauer Postfach 1171 C, 8458 Sulzbach ☎ 0 96 61/[illegible] bis 21 Uhr

● Lohn- und Einkommensteuer ●
Super Jahresausgleich + Steuererklärung + Rentenwertung + Analyse. Atari Sinclair Commodore Jahr. Aktualisierung 10,– DM Case 60,– Disk 70,– DM. Info Horst [illegible]chan, Niederfelder Str. 44, 8072 Manching ☎ 0 84 59/16 [illegible]

Atari 600 XL/800 XL/130 XE-User!
Lichtgriffel mit Programmen und deutscher Anleitung nur 44,– DM. Versand gegen Scheck/Nachnahme. Info gratis! Firma Schößauer Postfach 1171 A, 8458 Sulzbach ☎ 0 96 61/85 92 bis 21.00 Uhr.

●●●● Sinclair QL ●●●●
Hard- und Software
Z.B. extrem schnelle QL-RAM Disk DM 59,– 512 K-Erweiterung mit durchgeführtem Bus DM 374,– DFÜ Modapter DM 129,– Druckerinterface DM 89,– Über 100 Programme ab DM 25,– Disketten Upgrades nur DM 56,– usw. Sofort Info anfordern bei Firma Uwe Fischer Postfach 10 21 21 2000 Hamburg 1 ☎ 0 40/ [illegible] 04 61

● Lohn- und Einkommensteuer ●
Super Jahresausgleich + Steuererkl. wahl + Rentenwertung + Analyse Atari Sinclair Commodore Jahr. Aktualisierung 10,– DM Case 60,– Disk 70,– DM. Info Horst [illegible]chan, Niederfelder Str. 44, 8072 Manching ☎ 0 84 59/16 [illegible]

● Disketten ●
● 5¼" 48 tpi DM 0,95 2D ●
● 3½" 135 tpi DM 3,18 1DD ●
● 3" Schneider DM 5,85 ●
● Auch andere mit Garantie ●
● Allg. Austro-AG Ringstr. 10 ●
● D-8057 Eching ☎ 0 81 33/61 10 ●

NEU!!

Der Katalog Nr. 6

auf 112 Seiten Hardware, Software und Bücher
für AMSTRAD/SCHNEIDER CPC 464 664 6128 JOYCE
COMMODORE 64 UND 128, SPECTRUM/SPECTRUM PLUS

Katalog gegen 1,40 DM in Briefmarken anfordern bei

THOMAS WAGNER
SOFTWAREVERSAND

T. WAGNER SOFTWAREVERSAND P.O. BOX 112243 D-8900 AUGSBURG 1

unicorn soft
U&J Rudolph
Versand
Ehlenerstr. 7
3501 HOOF
Verkauf
Druselturm
3500 KASSEL
0561/770367

ATARI ST Tagestiefstpreis	
Spectrum	ab 248,–
Sinclair QL	ab 446,–
Tastatur f. QL	ab 188,–
Opus Discovery	ab 349,–
Seikosha SP 1000	ab 689,–

Computer, Drucker + Zubehör in unserer Gratisliste
Unseren Kunden
ein frohes Weihnachtsfest
Ursula & Joachim Rudolph

Bestellschein für private Kleinanzeigen

Bitte veröffentlichen Sie in der nächsten Ausgabe folgende Anzeige

Meine Anzeige soll in ☐ 1 ☐ 3 ☐ 10 Ausgaben erscheinen.

Rubrik

Vor- und Zuname

Straße

PLZ/Ort

Datum Unterschrift

Pro Zeile 1,– DM. Chiffre-Gebühr 5,– DM. Bei mehreren Kleinanzeigen bitte Bestellschein kopieren. Bitte den Betrag in Briefmarken oder als Scheck zusammen mit der Kleinanzeige einsenden an:

**Computer Kontakt
Redaktion
Postfach 16 40
7518 Bretten**

●●●● Das 1050 Turbo-Modul ●●●●
● Echtes Double Density 0000 01 ●
● Turbodrive Backup-Funktionen ●
● Druckerinterface nur 9[illegible],– DM! ●
● Druckerkabel dazu nur 4[illegible],– DM! ●
● Test CK 8-9/86 Garantie bei ●
● Bernhard Engl Steinstr. 13 ●
● 8000 München 80 ●
●●●●●●●●●●●●●●●●●●●●●●

★ Spectrum ★
★ Software ★

Datenmanager 84	DM 69.90
Hardcopy für Discovery	DM 29.90
Masken-INPUT	DM 39.00
Lagerverwaltung	DM 69.00
Faktura	DM 99.00
Inventur	ab DM 38.00
Discmenu	DM 38.00
Cartridgemenu	DM 28.60
Tasword-II Ergänzung	DM 68.00

**Kai Uffenkamp
Soft- und Hardware
Gartenstraße 3
4904 Enger
☎ 05224/2375**

FUNDGRUBE

Ergänzung zur RAM-Disk von Michael Gehret

In Heft 8-9/86 hatten wir auf Seite 95 die RAM-Disk von Michael Gehret vorgestellt. Dazu teilte uns der Autor einige Neuerungen mit. In der jetzt vorliegenden Version 2.0, die insgesamt als 1-Megabyte-Laufwerk simuliert werden kann, können TODISK-Ordnern und auch Unterordner, also komplette Anwendungen kopiert. Schreib- und Leseoperationen werden auf dem Bildschirm angezeigt. Das Programm funktioniert auch mit Festplatten und kostet rund 100,– DM. Wer die alte Diskette einschickt, erhält gegen 10,– DM ein Update.

Weitere Informationen:
Michael Gehret Uniware
Silcherstraße 15
8944 Grönenbach

Pil-Software Darmstadt

Unser Club besteht schon seit über einem Jahr. Schwerpunkte unserer Arbeit sind: Hilfestellung für Anfänger, Umgang mit Dienst- und Anwenderprogrammen, Erlernen und Programmieren verschiedener Computersprachen und anderes mehr. Hierzu bieten wir beispielsweise folgende Leistungen: regelmäßige Treffen, Betreuung durch unsere Systemsprecher, Software-Bibliothek, PDS, Club-Zeitung, Zeitschriften-Abos und Fachliteratur.

Der Club-Beitrag wird jährlich entrichtet und ist nach Altersgruppen gestaffelt. Mitmachen können Computer-Freaks mit allen Rechnern.

Wir suchen noch Mitglieder im gesamten Bundesgebiet und im Ausland, um weitere Bezirksgruppen gründen zu können. Infos sind gegen Rückporto beim Club erhältlich.

Peter Schuh
Pil-Software Darmstadt
[illegible] 25
6100 Darmstadt
Tel. [illegible]

Neue Pascal-Version

Von Kyan-Pascal (wir berichteten in Ausgabe 6/7) ist jetzt die überarbeitete Version 2.0 erhältlich. Auffälligste Neuerung ist die sog. KIX-Benutzeroberfläche, die den Umgang mit Editor und Compiler vereinfacht, da nicht in das DOS zurückgesprungen werden muß.

Weiterhin gibt es nun zum Kyan-Pascal zwei Toolkit-Disketten mit einer Menge nützlicher Pascal-Routinen. Das erste Toolkit enthält System-Utilities, das zweite nennt sich "Atari Advanced Graphics" und enthält Routinen zur 3-D-Transformation sowie zur PM-Grafik. Wir werden in einer der nächsten Ausgaben noch genauer darüber berichten.

Weitere Informationen:
Compy Shop
Gneisenaustr. 29
4330 Mülheim

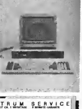
Peter Fink

Ein neuer QL?

Aus England erreicht uns die Meldung, daß ein neuer Computer mit der Bezeichnung Thor auf den Markt kommen soll. Das allein ist natürlich noch keine besondere Nachricht. Für Sinclair-, speziell für QL-Benutzer ist aber interessant, daß er voll QL-kompatibel sein soll. Gerüchten zufolge soll in Thor die Original-QL-Platine stecken, die nur leicht modifiziert wurde. Leider war es noch nicht möglich, ein Testgerät zu erhalten. Wir bemühen uns aber intensiv darum und hoffen, es bis zur nächsten Ausgabe zu schaffen. Vorab schon mal ein Foto des neuen Computers und einige technische Daten:

- 640 KByte RAM
- eingebautes 3,5-Zoll-Floppy, optional 20-MByte-Harddisk
- Steckplatz für User-EPROM
- Echtzeituhr
- IBM-ähnliche Tastatur
- Schnittstellen für Maus, Centronics- oder serielle Drucker usw.

Rolf Knorr



Dieses Ding ist voll QL-kompatibel und heißt Thor

SPECTRUM • QL • ATARI • SPECTRUM • QL • ATARI • SPECTRUM

H.G. Dreeser, Soft- und Hardware

Wir bekommen laufend Neuheiten an Produkten für den Spectrum, QL, Atari 800/800/130 XL/XE sowie Atari ST. Nutzen Sie unseren Zu[illegible] Auftragsservice [illegible]

Spectrum		QL	
Lightforce	34,50 DM	Mortville Manor	[illegible]
Great Escape	34,50 DM	Nucleon	[illegible]
Druid	34,50 DM	Othello	49,90 DM
Cobra	34,50 DM	Vroom	49,90 DM
Infiltrator	[illegible]	Westerner 3D	62,90 DM
Dan Dare	[illegible]	Toolkit MK II	147,00 DM
Dragons Lair	34,50 DM	Macro Assembler	[illegible]
Spectrum 128		**Hardware**	
[illegible]	39,90 DM	AMX Mouse (Spectr.)	[illegible]
Ghostbusters	39,90 DM	QL [illegible]	[illegible]
[illegible]	39,90 DM	Spectrum 128 (D)	399,00 DM
Nodes of Yesod	39,90 DM	Discovery one/48	[illegible]
Planets	39,90 DM	Discovery one/128	[illegible]
[illegible] World	[illegible]	Multiface One	[illegible]
Gladiator	[illegible]	Cartridges	[illegible]

Fordern Sie unsere Gratisliste an
Dreeser Soft- u. Hardware, [illegible]
[illegible]

SPECTRUM • QL • ATARI • SPECTRUM • QL • ATARI • SPECTRUM

SPECTRUM SERVICE
REPARATURZEIT CA. 1 WERKTAGE 3 MONATE GARANTIE
KOSTENVORANSCHLAG AUF WUNSCH (KOSTENLOS)

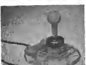
MULTICARD 1 0

16 BIT SIGNALAUSGABE PORT
EPROM KARTE
EPROM PROGRAMMIERGERAET

3 GERAETE IN EINEM [illegible] ANWENDUNGSMOEGLICHKEITEN
FORDERN SIE VON UNS WEITERE INFORMATIONEN AN

SUPERPREIS NUR 169,00 DM incl. MwSt

BELKENHEIR COMPUTERTECHNIK
WELLINGER WEG [illegible]
TEL [illegible]



Der Competition Pro gewährt Einblicke auf Schalter

Joystick in durchsichtigem Kunststoffgehäuse

Der Competition-Pro-Joystick ist robust, schnell und praktisch; Spiele-Fans wissen das. Neu am Competition Pro ist jetzt das transparente Kunststoffgehäuse. Damit ist Joystick-Technik sichtbar gemacht, und man kann während des Spielens sehen, wie's funktioniert. Natürlich ist auch der neue Joystick mit Mikroschaltern und 8-Wege-Steuerung ausgestattet.

Weitere Informationen
Dynamics GmbH
Friedensallee 35
2000 Hamburg 50

Soundbox für C 64/C 128

Für die User von C 64 und C 128 mit einem Monitor ohne Lautsprecher ist diese Erweiterung interessant. Die Soundbox stellt einen kompakten, preiswerten HiFi-Verstärker mit eingebautem Lautsprecher dar, der an der Audio-Video-Buchse des Computers angeschlossen wird. Auch der Monitor arbeitet nach wie vor an diesem Anschluß (bzw. der Antennenbuchse), da alle Leitungen durchgeschleift werden. Die Stromversorgung erfolgt über vier Mignon-Batterien, der Anschluß eines Netzgerätes ist nicht vorgesehen. Die Lautstärke wird links am Gerät geregelt, eine rote LED dient als Funktionsanzeiger. Das Potentiometer arbeitet leider nicht ganz linear, so daß man bei höheren Lautstärken ein ganz schönes Fingerspitzengefühl braucht.

Es ist schon erstaunlich, was die circa 5x9x5 cm große Box an Sound zu bieten hat. Nur bei tieferen Baßtönen hat sie ihre Probleme, was bei solch geringen Abmessungen nicht verwundert.

Die Soundbox ist zwar keine Alternative zum Anschluß des Computers an die Stereo-Anlage, stellt aber sicher eine sinnvolle Ergänzung zum Monitor ohne Lautsprecher dar.

System C 64/C 128
Hersteller
Wiesemann & Theis GmbH
Preis ca. 78,– DM

Stefan Seibach

Ein geschmackvolles Design rundet den positiven Eindruck der Soundbox ab

Star-Painter von Sybex

Achtung: Der "Star-Painter" von Sybex kostet nicht wie irrtümlich im letzten Heft angegeben 79 DM, sondern nur 64 DM. Wir bitten dieses Versehen zu entschuldigen.

Inserentenverzeichnis

Alfa	S. 108
AMC Verlag	S. 6
Annette Länge	S. 9
Bollenhaus	S. 121
C & M Mayer	S. 17
Compy Shop	S. 71
CSV Riegert	S. 3
Dreesen	S. 57/123
Elektor Verlag	S. 28
Flensburger Computerversand	S. 117
Holschuh	S. 8
Hüthig Verlag	S. 7
Individual Software	S. 25
Irata	S. 87
Jäkel & Kästwerth	S. 41
Jeposoft	S. 5
Joysoft	S. 119
Jupitersoft	S. 3/17
Jürgen Dörr	S. 65/79
Klemmer + Schulte	S. 44
Kurz	S. 6
Krawietz	S. 116
Lucker	S. 6
Naujoks	S. 5
Nawi's	S. 9
Rausch + Haub	S. 101
Rätz-Eberle	S. 124
Schillhauser	S. 120
Teia Verlag	S. 2
T.S. Datensysteme	S. 123
Uffenkamp	S. 110
Unicom	S. 3/17 120
Wagner	S. 120

Fundgrube

Wer mit einem 3- oder 3,5"-Laufwerk arbeitet und bisher nicht wußte, wohin mit den teuren Disketten, sollte sich einmal die neue Multiform-Diskettenbox ansehen. Im Gegensatz zu herkömmlichen Boxen ist sie zusammenklappbar und benötigt nur relativ wenig Platz; sie hat dann ungefähr das Format eines mittelstarken Buches. Allerdings können auch nur 12 Disketten untergebracht werden, was für viele Anwender aber ausreichen dürfte.

Weitere Informationen
Jeposoft
Kruppstraße 9
4040 Neuss 21

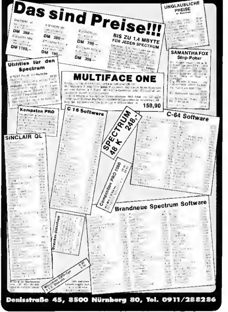
Impressum

Herausgeber: Dipl.-Wirt.-Ing. (FH) Thomas Eberle, Werner Rätz
Chefredakteur: Thomas Eberle
Technische Redaktion: Werner Rätz
Redaktion: Helmut Fischer, Robert Kaltenbrunn
Ständige freie Mitarbeiter: Horst Müller, Rolf Knorre, Dipl.-Ing. H. P. [illegible], [illegible], Michael [illegible], Rainer W. Gerling, Christian [illegible], Dipl.-Ing. Peter [illegible], Thomas [illegible], Karlheinz [illegible], Martin [illegible]
Verantwortlich: Gabriele Herzog
Titelbild: [illegible]
Anzeigen: Anna Weiß
Es gilt die Anzeigenpreisliste Nr. 5 vom 1.1.1986
Montage: Véronique [illegible]
Satz: [illegible], 7143 Vaihingen/Enz
Druck: Verlags- und Industriedruck GmbH & Co KG, 7730 Villingen-Schwenningen
Vertrieb: Verlagsunion, 6200 Wiesbaden
Anschrift des Verlags: Verlag Rätz-Eberle, Postfach 1640, [illegible] 71, 7518 Bretten, Tel. [illegible]

Manuskript- und Programmeinsendungen: [illegible]

Computer Kontakt erscheint zweimonatlich am letzten Montag des Vormonats und kostet pro Heft [illegible] DM

T.S. Datensysteme-Vertriebsgesellschaft mbH



Das sind Preise!!!

DM 399,–
DM 599,–
DM 799,–
DM 1199,–
DM 199,–
DM 399,–

BIS ZU 1,4 MBYTE
FÜR JEDEN SPECTRUM

UNGLAUBLICHE
PREISE

SAMANTHA FOX
Strip-Poker

Utilities für den Spectrum

MULTIFACE ONE

159,90

Kempston PRO

C 16 Software

SPECTRUM
48 K 248,–

C-64 Software

SINCLAIR QL

Competition PRO 5000

Brandneue Spectrum Software

Drucker-Interface

Microdrive Cartridge
3 1/2 Diskette

Denisstraße 45, 8500 Nürnberg 80, Tel. 0911/288286

ATARI magazin
Das unabhängige Magazin für alle Ataris
Neu: Ab dem 22.12. bei Ihrem Zeitschriftenhändler!
Tips für die 1050-Floppy mit Happy-Enhancemen